drones

MDPI

Article

Oxpecker: A Tethered UAV for Inspection of Stone-Mine Pillars

Bernardo Martinez Rocamora, Jr. †, Rogério R. Lima †, Kieren Samarakoon, Jeremy Rathjen, Jason N. Gross and Guilherme A. S. Pereira *

Department of Mechanical and Aerospace Engineering, Benjamin M. Statler College of Engineering and Mineral Resources, West Virginia University, Morgantown, WV 26506, USA
* Correspondence: guilherme.pereira@mail.wvu.edu
† These authors contributed equally to this work.

Abstract: This paper presents a state-of-the-art tethered unmanned aerial vehicle (TUAV) for structural integrity assessment of underground stone mine pillars. The TUAV, powered by its tether, works in tandem with an unmanned ground vehicle (UGV) that hosts the TUAV batteries, a self-leveled landing platform, and the tether management system. The UGV and the TUAV were named Rhino and Oxpecker, respectively, given that the TUAV stays landed on the UGV while the ensemble moves inside a mine. The mission of Oxpecker is to create, using a LiDAR sensor, 3D maps of the mine pillars to support time-lapse hazard mapping and time-dependent pillar degradation analysis. Given the height of the pillars (7–12 m), this task cannot be executed by Rhino alone. This paper describes the drone's hardware and software. The hardware includes the tether management system, designed to control the tension of the tether, and the tether perception system, which provides information that can be used for localization and landing in global navigation satellite systems (GNSS)-denied environments. The vehicle's software is based on a state machine that controls the several phases of a mission (i.e., takeoff, inspection, and landing) by coordinating drone motion with the tethering system. The paper also describes and evaluates our approach for tether-based landing and autonomous 3D mapping of pillars. We show experiments that illustrate and validate our system in laboratories and underground mines.

Keywords: aerial robotics; mine inspection; tethered unmanned aerial vehicle; tethered drone; autonomous landing; tether-guided landing

check for updates

Citation: Martinez Rocamora, B., Jr.; Lima, R.R.; Samarakoon, K.; Rathjen, J.; Gross, J.N.; Pereira, G.A.S. Oxpecker: A Tethered UAV for Inspection of Stone-Mine Pillars. *Drones* **2023**, *7*, 73. https://doi.org/10.3390/drones7020073

Academic Editors: Sophie F. Armanini and Raphael Zufferey

Received: 16 December 2022
Revised: 13 January 2023
Accepted: 17 January 2023
Published: 19 January 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Unmanned aerial vehicles, and more specifically multi-rotor drones, are progressively being used in the mining industry [1]. They are used for surface and underground mines, in operation or abandoned [2]. In surface mines, where Global Navigation Satellite Systems (GNSS) are available, several applications, such as 3D mapping [3,4], stockpile volumetric estimation [5], slope instability detection [5,6], erosion assessment [7], and site monitoring [8], can be performed with off-the-shelf drones. However, for underground applications that include geotechnical characterization [9,10], gas detection [11], and search and rescue missions [12], adaptations to the vehicles and their navigation software need to be made. The missions above, which are often executed by human technicians, can be carried out by autonomous or semi-autonomous drones that, due to their increased speed and range, can positively affect the efficiency, productivity, and safety of the task.

One of the main challenges for environments such as underground mines is the lack of access to GNSS [13,14]. Teleoperation is also a challenge due to the limited range of radios underground. Therefore, a robust onboard navigation system, which includes localization, mapping, and motion planning, must be in place for a fully autonomous system to function reliably. Whereas visual-based solutions, including visual inertial odometry (VIO), can mostly solve the localization and mapping in general GNSS-denied environments [15,16], underground environments are often dark and lack distinct features [17], motivating

modern solutions to rely on light detection and ranging (LiDAR) sensors [18] and thermal cameras [19]. An additional drone limitation emphasized underground is its short flight time, which limits the area that can be covered in a single mission.

This article describes the development of a state-of-the-art tether-powered unmanned aerial vehicle (TUAV), which is part of the complete autonomous robotic system shown in Figure 1. The system, composed of an unmanned ground vehicle (UGV), called Rhino, and the TUAV, known as Oxpecker, was designed to provide early warning signs on underground stone mine pillar structural integrity. When in operation, the system records 3D point-cloud data of pillars to support time-lapse hazard mapping and time-dependent pillar degradation analysis. Because stone mines have pillars varying from 7 to 12 m in height, sensors on the UGV alone cannot obtain high-resolution data of the entire pillar, thus justifying the deployment of the TUAV each time a pillar must be surveyed. As the UAV batteries are carried by the UGV and the tether provides relative measurements between the vehicles, the proposed configuration not only increases the battery capacity of the drone but also allows drone navigation and landing without vision.

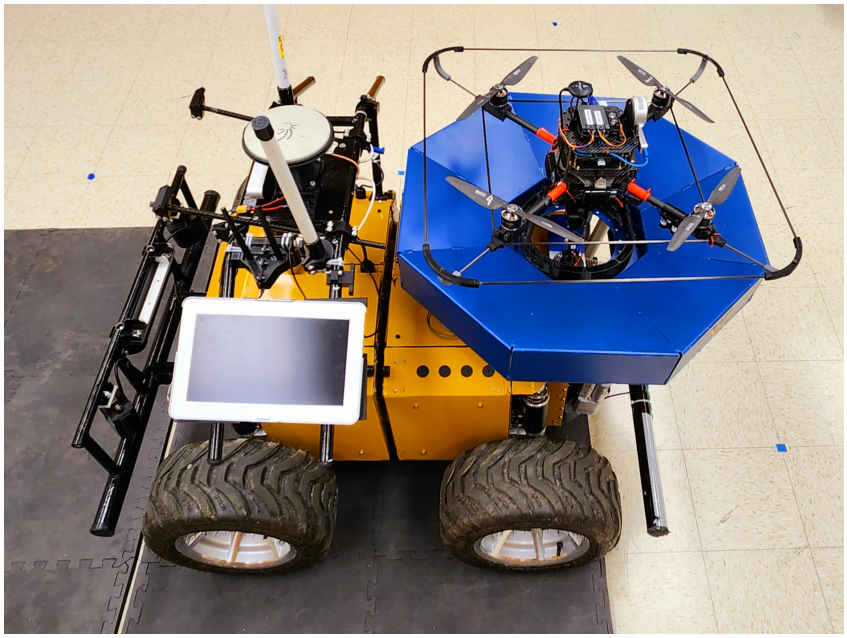


Figure 1. Rhino, the unmanned ground vehicle (UGV), is a mobile robot in yellow and blue. Oxpecker, the tethered unmanned aerial vehicle (TUAV), is the drone in black on top of the blue platform. Rhino and Oxpecker form a cooperative robotic system for underground mine mapping.

In our system, a set of high-energy density batteries was added to Rhino's payload to achieve long-endurance flights of Oxpecker. The batteries can deliver an average power of 570 W (i.e., 38 V at 15 A) to the TUAV through a power cable. Two other subsystems were designed to integrate and handle the tether. First, a tether management subsystem was developed to control the tension (i.e., force) on the cable, keeping it in suspension and preventing it from touching the ground. A second subsystem consists of a set of sensors used by the drone for localization and landing. These sensors measure the length of the tether and tether angles at both endpoints (i.e., the TUAV and the landing platform on the UGV). This paper also describes the software architecture that handles the different phases of the TUAV mission: autonomous takeoff, mapping, and landing. These phases are driven by a state machine, which communicates with the tethering system via robot operating system (ROS) [20] messages over a local WiFi network. For the mapping phase, we developed a path planning methodology to create a dense point-cloud map of the pillars while avoiding collision with the floor and ceiling and a LiDAR-based strategy for controlling the TUAV. We also introduce a robust landing approach that does not depend on any external sensor, such as vision or LiDAR, and only relies on measurements from the

tether angle sensor (azimuth and elevation angles) and altitude readings coming from an altimeter. This approach is able to safely and robustly make Oxpecker approach and land on the platform on Rhino.

Therefore, the contributions of this work are:

1. The design of a tether-powered multi-rotor UAV for long-term robotics,
2. The design of a tethering perception system,
3. The design of a cable management system capable of conveying power to the TUAV and also controlling the tension of the tether,
4. A motion planning and control strategy for unknown pillar surveying and mapping in a GNSS-degraded environment,
5. A strategy for autonomous landing using tether variables.

The rest of this paper is structured as follows. Relevant related work is presented in Section 2. The hardware of the UAV and companion systems is explained in Section 3. The software architecture is detailed in Section 4. Section 5 presents a series of field experiments that illustrate and evaluate the proposed approach. Finally, Section 6 presents conclusions and proposals for future work.

2. Related Work

Ground and aerial robots have been used separately for search and rescue and mapping applications in underground mines [10,14,21]. However, using robots of a single locomotion type introduces problems such as limited endurance and payload (in the case of an aerial robot) or restrictions on which places can be reached or inspected (in the case of a ground robot). Thus, recent studies show increased attention to multi-robot collaboration with teams of heterogeneous robots [12,22].

Six of the eight teams competing in the systems track of the Defense Advanced Research Projects Agency (DARPA) Subterranean Challenge (SubT), a game-changer competition for robotic technologies in subterranean environments [23–28], have used a combination of robots that included both UGVs and UAVs to map, navigate, and search underground environments during time-sensitive combat operations or disaster response scenarios. Team CERBERUS used legged, wheeled, and flying robots [23]. Team CSIRO Data61 used tracked vehicles, legged robots, and drones [24]. Team Explorer used custom-built wheeled and flying robots, some of them deployed by the UGVs [25]. Team CoSTAR used legged, wheeled, and drones [26]. Team CTU-CRAS-NORLAB used legged, tracked, and flying robots [27]. Team Coordinated Robotics used wheeled and flying robots [23]. Only teams Robotika, which used wheeled robots, and team MARBLE [28], which used legged and wheeled robots, avoided the use of aerial robots [23].

During the SubT competition, all teams that used aerial robots, including the winning team [23], deployed the drones independently from the ground robots. While this permitted the teams to increase the reach of their systems, they still faced the problem of endurance, as the flight time for battery-powered drones is very short. For example, the RMF-Owl [29], a collision-tolerant aerial robot developed by team CERBERUS, is equipped with a 5000 mAh LiPo battery and is only able to fly for 10 min. Similarly, Flyability's Elios 3 [30], a state-of-art commercial drone designed for underground mining inspection and whose predecessor, Elios 1, was also used by team CERBERUS, is only able to fly for 12.5 min (without payload). More recently, in [31], new developments are trying to increase the time of operation of a drone by carrying it on the back of a quadruped robot, which would allow continuous operation for about an hour. However, without any recharging strategy, the drone flight time is still limited to the same 10 min.

In our work, we propose to have a ground vehicle carrying a drone and a tether connecting both robots. The tether is used to power the UAV with large batteries carried by the UGV. The obvious benefit of including a tether is an extended flight time for the UAV, given that batteries represent a big share of the takeoff weight of a drone. This solution has been explored before in [32]. A comparison of the energy consumption between battery-powered and tether-powered drones was provided by [33]. More recently, research on

providing power for a series of quadcopters showed that the serialization of tethered drones could make the system even more efficient and advantageous [34]. However, although tethering the drone can increase flight time, it comes at the cost of reduced reach. In our case, this is compensated for by the mobility of the ground robot and the nature of the environment where the robotic system is deployed, i.e., cavernous and of limited height. A complete review of alternatives to power drones, which includes complex tasks such as swapping batteries [35] or self-recharging [36], is presented in [37]. Although we considered some of these alternatives in our work, the additional benefits brought by the tether, including precise landing and localization [38], led us to decide on this approach.

If properly instrumented, the tether connecting the UAV to its base (in our case, a UGV) can provide relative localization [39]. The approach developed by our research team is described in [38], which was followed by a similar approach in [40]. The main advantage of these works with respect to previous ones [41,42] is the fact that they do not assume that the tether is taut or has minimum slack [43], but, instead, allow it to be slack by modeling its shape with a catenary curve.

Finally, assuming instrumentation similar to that used for tether-based localization, the tether can also be used to aid in landing the UAV. In contrast to previous landing methodologies [44–47], this allows the drone to land precisely without the requirement of the relative localization between the vehicle and the landing platform, which can be noisy and biased. Precise landing using a tether to guide the UAV towards the landing point was explored in [48–50].

The next section will describe the hardware of our system, which includes all the sensors used for localization and landing.

3. System Hardware

This section describes the design requirements and engineering decisions made by our team for the TUAV system, which includes the tether management and sensing system and an actuated landing platform.

3.1. Mission Requirements and Design Decisions

The main requirement of this project is that the pillars of the mine, the height of which may vary from 7 to 12 m, are autonomously and completely scanned with a LiDAR or camera so that a 3D map can be constructed and used for hazard assessment. Although a standard UAV could be applied to this task, its short flight time would not be practical for mines with hundreds of pillars. On the other hand, a UGV alone could not scan a single pillar completely unless it is equipped with a telescopic device/manipulator that would be able to move the sensors along the pillar until its top. After considering this solution as cumbersome and costly, our team then considered the use of a drone, which would work in tandem with the UGV. The mission requirements for this drone are listed as follows:

- Able to operate for several hours without human intervention;
- Able to carry mapping and surveying sensors, including LiDARs and cameras;
- Able to precisely land on the UGV before moving from one pillar to the next.

These requirements lead the research team to design:

- A tether-powered quadrotor with enough payload capacity to carry localization and 3D mapping sensors, and the weight of the released power cable;
- A tethering system that manages the cable release and retraction and that assists the drone localization and landing by measuring relevant variables, such as tether angles and tether length;
- A self-leveling landing platform that compensates for the roll and pitch of the ground vehicle and assists in drone landing.

The development of the drone and the other subsystems is described in the following subsections.

3.2. TUAV System

This section overviews the design concept of Oxpecker, the proposed TUAV. The goal of this design was to develop a drone that is able to be powered by a cable and has enough payload capacity to carry the required instruments and the tether. The main challenge of designing such a system was working on the trade-offs of transmitting such high power over a long cable. Our solution required a combined selection of frame, powertrain, and cable. To complete the design, we selected the flight controller, companion computer, and sensor suite onboard the drone. A schematic overview of the hardware setup can be seen in Figure 2. Table A1 lists the components required to assemble Oxpecker along with their manufacturers, models, and other relevant specifications. The most relevant components are discussed below.

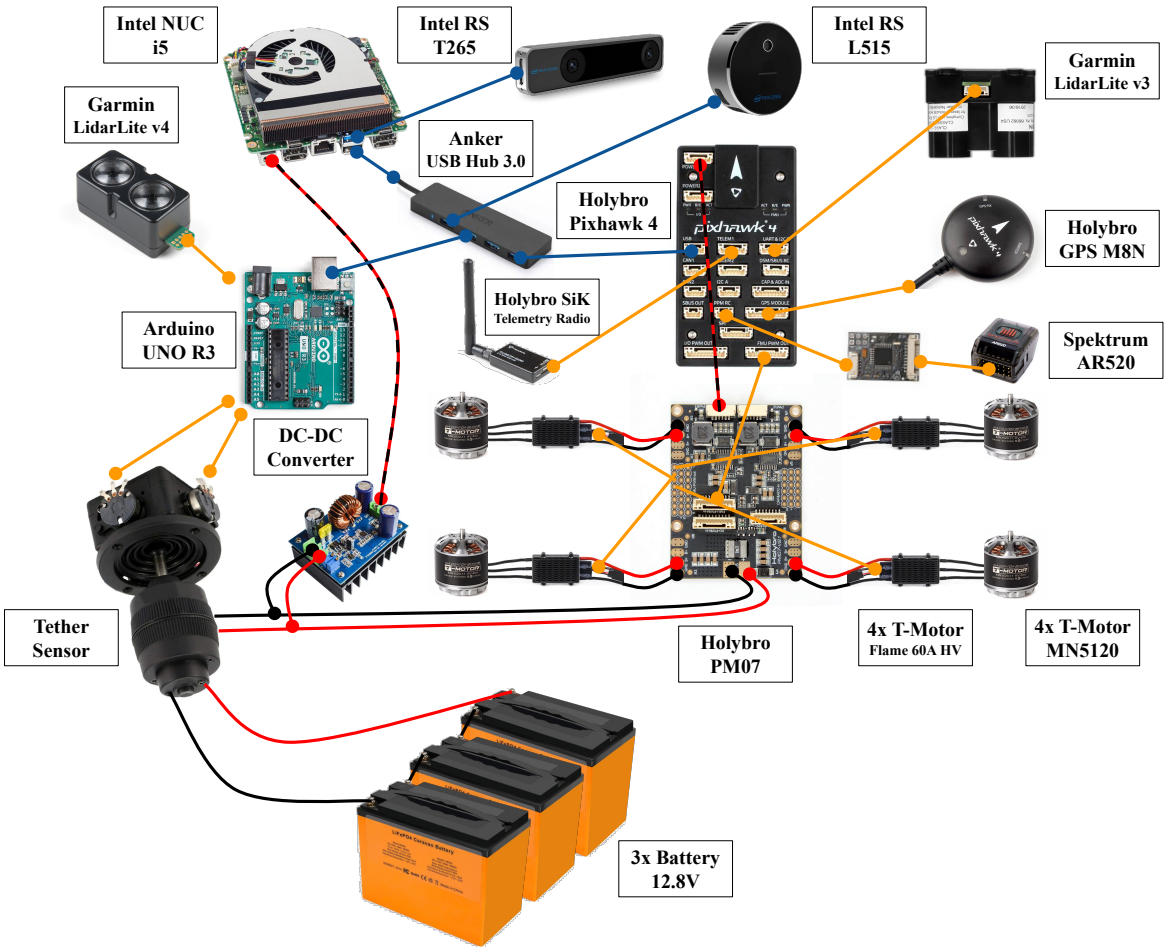


Figure 2. Schematic overview of Oxpecker's electronics. Except for the battery pack, all other components are carried onboard by the drone.

3.2.1. Cable Specification

An electrical cable has its resistance given by $R = \rho\, L/A$, where L is the cable length, ρ is the material resistivity, and A is the cross-sectional area. Therefore, the longer the cable, the higher the resistance. In the presence of a current i, the voltage drop Δv along the cable is ruled by $\Delta v = Ri$. The current and voltage drop directly translate into energy loss by Joule's law (heat dissipation), which states that the power converted from electrical energy to thermal energy is given by $P = \Delta v \cdot i = (\rho L/A) \cdot i^2$. One way to reduce the resistance, and thus the power loss, is to increase the cross-sectional area of the conductor. However, this approach contributes to increasing the weight of the cable to be suspended by the drone. We could also reduce the length of the cable, which would limit the reach of the drone. A feasible alternative to minimize the power loss without shortening the cable

is raising the voltage delivered to the drone, thus reducing the current flowing through the tether for the same power consumption. As the power loss on the cable scales with the square of the current, this would be the best approach to take. Additionally, smaller currents permit the use of conductors with reduced cross-sectional areas, which contributes to decreasing the cable's linear weight. Many commercial applications prefer to transmit power at a very high voltage (e.g., 400 V) and include a voltage converter on the UAV to reduce the voltage level required by the motors and flight controller. However, as the payload is a critical constraint in our design and power conversion devices are usually not lightweight, we decided to increase the voltage as much as possible to connect it directly to the power management board without a DC-DC converter.

Given an estimated power-to-weight ratio of ≈150 W kg^{-1}, which is consistent with the specific power estimates of 200 W kg^{-1} for agile quadrotors shown in [51], and a maximum hovering weight of ≈4 kg, we selected the cable for an expected power consumption of 600 W on the drone. We then chose to use three 12.8 V batteries connected in series to power the TUAV. Given the open-circuit battery voltage of $v_b = 3 \times 12.8 = 38.4$ V and the required power of 600 W, an initial current estimate on the cable without assuming any voltage drop would be ≈ 15.6 A. By assuming a 16 AWG stranded copper cable with resistivity 0.0132 Ω m^{-1}, which has adequate flexibility for our application, and also considering the additional resistance of 0.15 Ω from connectors and other parts of the circuit, we can iterate the voltage drop until convergence is reached. By applying this process, we compared different tether lengths and opted for a 15 m-long cable, which is sufficient for our task. The estimated voltage drop in this cable is $\Delta v \approx 6.6$ V when the current is 18.8 A. This yields a power loss of ≈123.5 W along the tether.

The powertrain of the quadcopter, specifically, electronic speed controllers (ESCs), motors, and propellers, were chosen to handle high voltage (38.4 V) from the tether and provide enough thrust to match the maximum hovering weight of the drone (≈4 kg). We considered that the system would work around the 8S voltage rating after accounting for the losses on the cable and expected power consumption for the drone (≈600 W). The power delivered was considered to be approximately constant during its mission, as the drone is not expected to be agile and will work in near-hover flight conditions all the time. The selected motor was the T-MOTOR MN3520 (rated 4S-8S) with a T-MOTOR 12 × 4 carbon fiber propeller, using a T-MOTOR FLAME 60A HV ESC (rated for 6S-12S). In our tests, we measured the voltage (≈33.5 V) and current (≈18.5 A) delivered to Oxpecker during flight, confirming the expected power consumption (≈620 W) and validating the selection of the powertrain.

3.2.2. Frame Selection

The frame is the structure that holds all the components of a quadrotor together. It needs to be designed to be strong but also lightweight. The lighter a frame is, the more payload the drone can carry. A great way to optimize the airframe is to select carbon fiber frames. We chose an easy-access, commercial frame for this aerial system, the Holybro X500 Frame Kit. This frame was selected because it is easy to assemble and is capable of supporting the estimated maximum takeoff weight for this drone (4 kg). Adaptations have been made to install the heavier, higher voltage motors selected (e.g., 3D printing reinforced clamps that hold the arms and the motors).

3.2.3. Flight Controller Selection

A review of flight controller options in [52], shows that, typically, research groups rely on Pixhawk-PX4, Parrot, or DJI low-level controllers. They are mostly used as black boxes, but they differ in how much authority is given to change the controllers. The flight controller unit (FCU) selected for the drone was the Pixhawk 4, which is optimized to run PX4, an open-source flight control software for UAVs and other unmanned vehicles. It provides all the main sensing capabilities required by a UAV, such as an inertial measurement unit (IMU) and barometer, and ports to integrate several others directly. For Oxpecker, we

integrated a U-Blox Neo-M8N GPS/GLONASS receiver to provide a compass, a SiK Telemetry Radio V3 for monitoring and debugging purposes using QGroundControl, a Garmin LidarLite v3 ranging sensor to provide height measurements, and a Spektrum AR620 receiver to provide manual radio control. A large community also supports the Pixhawk, and it can easily be integrated with a companion computer through the MAVLINK [53] communication protocol.

3.2.4. Companion Computer Selection

Given that the Pixhawk 4 has very limited onboard computing capability, a companion computer is required to run custom software for the drone and allow for data collection (i.e., images and point clouds). There are many options for companion computers, and the most popular are listed in [52]. When no GPU is required, the Intel NUC family, the Udoo family, and the Raspberry family of computers are the most popular options. From our previous experience, fully autonomous operation, including localization and mapping, is possible with an embedded computer with extended RAM and storage. Originally, the Udoo X86 II Ultra with an Intel® Pentium® N3710 (2.56 GHz, 8 Gb RAM, 32 GB eMMC Storage, 150 g, 6 W) was tested due to its lower cost. Moreover, this computer also contains an Arduino Leonardo-compatible platform embedded on the same board, which facilitates the integration of I2C and analog sensors, such as the potentiometers we use to measure tether angles. However, although enough to perform basic functions (landing, wall following, etc.) and to save data, this computer was not powerful enough to run simultaneous localization and mapping (SLAM) algorithms online. Subsequently, we opted to substitute this embedded computer with an Intel® NUC 11 with an Intel® Core™ i5-1135G7 processor (8M cache, up to 4.20 GHz) and an external Arduino UNO R3 for low-level sensor communication. Additionally, we included a 2TB SSD card for storing the mapping data. The custom software running in this computer was developed using the Robot Operating System (ROS) and is described further in this paper. The communication between the companion computer and the Pixhawk was done using the MAVROS package [54]. A USB hub was connected to the computer to allow for several USB devices: WiFi antenna, notification light, Arduino, Pixhawk, stereo camera, and LiDAR.

3.2.5. Sensors Suite Selection

Localization in a GNSS-denied environment is a challenging problem [19], and redundancy is required. In our design, the sensor arrangement converged to ranging and distance sensors, a tether angle sensor, and a tracking camera for visual inertial odometry (VIO). A telemetry antenna, a radio controller receiver, and a GPS antenna (here for the compass) were also added directly to the flight controller. Some of these sensors are described next.

Ranging Sensors

Given the environment where the system is deployed, range measurements to the floor and ceiling are crucial for safe flight. In our design, these distances are measured using Garmin LidarLite optical distance measurement sensors, which are compact, lightweight, and have low power consumption. A LidarLite v3, measuring the distance to the floor, was integrated directly into the flight controller (Pixhawk 4), being used on its internal EKF and for the landing procedure. A LidarLite v4, measuring distance to the ceiling, was integrated into the TUAV using an Arduino UNO R3 to guarantee a safe distance from the mine ceiling. Additionally, a DWM1001-DEV development board, which includes the DWM1001C ultra-wideband (UWB) transceiver module, was added to the system to provide ranging measurements between the TUAV and the UGV. Although not discussed in this paper, UWB data can be used to improve the drone's relative localization in sensor fusion algorithms [55,56].

Tracking Camera

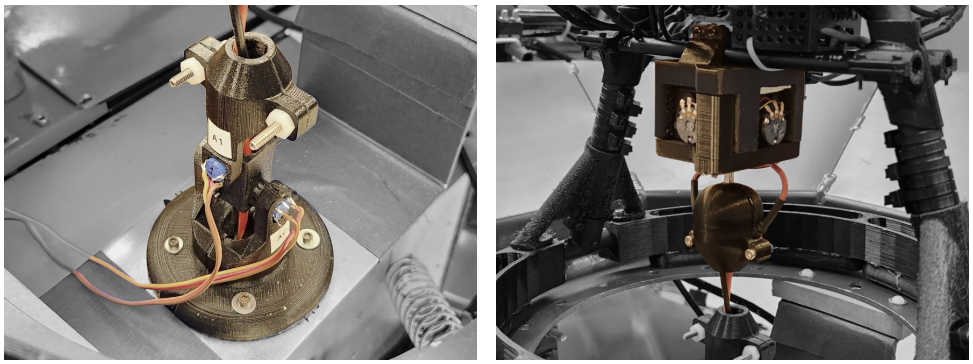
A requirement for autonomous flight is a good odometry solution. A complete solution for position and velocity estimation is the Intel® RealSense™ Tracking Camera T265. It is a stand-alone module that runs V-SLAM (visual simultaneous localization and mapping) algorithms directly on it and provides a visual inertial odometry (VIO) solution. It is also a compact and lightweight module that operates at lower power. In our TUAV, the tracking camera was mounted facing forward. The main limitation of the T265 camera is that it is generally limited to light levels higher than 15 lux, which is low but is usually more than what is available in an underground mine. To deal with this problem, we included two spotlights on the TUAV to provide illumination. The Intel® RealSense™ T265 odometry solution was integrated directly into the flight controller localization solution.

Mapping Sensor

For our application, we tested the Intel® RealSense™ D435 depth camera, the Microsoft® Kinect™ Azure development kit, and the Intel® RealSense™ L515 solid-state LiDAR depth camera. These options have been compared before for SLAM applications [57], but usually not under low light conditions. Considering the performance that we observed in dark environments (noise level and susceptibility to reflections), and other metrics such as power consumption, and weight trade-offs, the L515 camera was chosen as the most suitable our application. The L515 was designed to be used in indoor, controlled environments. Compared to the D435 camera, the L515 is slightly heavier (100 g against 72 g), it has a smaller range (9 m against 10 m) and it has same power consumption (3.5 W). However, it provides measurements with lower noise levels in dark environments. Compared to the Kinect™ Azure, the L515 is significantly lighter (100 g against 440 g), it has a larger range (9 m against 3 m), it consumes less power (3.5 W compared to 5.9 W), and it is also more robust to low light conditions and reflective surfaces.

Tether Sensors

Sensors on the drone and the tethering system are used to measure the tether variables. These sensors are the encoder, which is connected to the winch and measures the length of the tether, and the tether angle sensors, which are designed to measure the azimuth and elevation angles of the tether on both endpoints, i.e., at the landing platform and the TUAV. Figure 3a shows the custom 3D-printed sensor specifically designed for angle measurement at the platform, whereas Figure 3b shows a commercial-off-the-shelf joystick device converted to be used as the tether angle sensor at the TUAV side. A more detailed explanation of the functioning of these devices is provided in the following subsection.

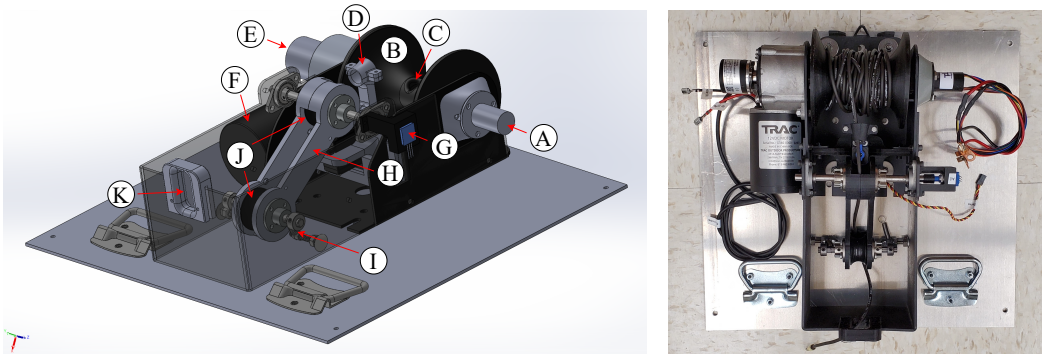


(**a**) At the UGV side. (**b**) At the TUAV side.

Figure 3. Tether angle sensors: (**a**) shows a kinematic chain with two degrees of freedom designed to measure azimuth and elevation angles at the platform; (**b**) shows an analog joystick adapted to obtain the same angles at the drone side.

3.3. Tethering System

The tethering system is composed of (1) a cable management system, responsible for controlling the tether tension by reeling the tether in or out, depending on the measured tension; (2) a tensiometer for measuring the tether tension; and (3) two tether angle sensors. Tension is measured by a mechanism composed of a pivoted spring-load lever arm with conveying pulleys where the tether slides off, as shown in Figure 4. A mechanism composed of a servomotor and a lever arm is used to evenly distribute the cable on the reel when turning, as depicted in Figure 4a(D). The lever arm moves synchronously with the reel's spin, which can be properly adjusted in firmware. The control system is accomplished by a digital proportional-integral-derivative (PID) controller implemented in an Arduino microcontroller. In order to deliver power to the TUAV through the cable, a slip ring is used to convey electrical power through the reel. The tethering system is assembled in a compartment located at the rear of the ground robot.



(**a**) CAD. (**b**) Hardware.

Figure 4. Cable management system. (**a**) Three-dimensional CAD model of the system highlighting the slip ring device (A), which allows the power to be transmitted from its external fixed frame to the internal rotating frame physically attached to a custom 3D printed spool (B) where the power cable is stored. The cable goes inside the spool's axis, comes out from (C), and passes through a funnel-like device (D). This device is driven by a servomotor that moves from side to side to evenly distribute the cable over the spool. The length of the tether is measured by a digital rotary encoder (E) when the DC motor (F) is turning. The tether tension is measured by a potentiometer (G) installed in the axis of a pivoted lever arm (H), which supports a pair of pulleys (J) where the cable can slide off. A spring (I) is used to load the lever arm. The cable access the system through the outlet (K). (**b**) Real-world implementation of the designed system.

Two tether angle sensors were designed in-house. A joystick-like device was adapted (see Figure 3b) for the sensor at the drone endpoint of the tether. Because the axes of rotation for this device are co-planar, the measurements of its potentiometers can be directly translated to the tether's azimuth and elevation angles by converting the voltage read by a microcontroller analog-to-digital converter (ADC) port and applying a calibration curve. A similar approach was used for the sensor at the platform endpoint of the tether, where we have the custom design shown in Figure 3a. To obtain the tether azimuth and elevation angles for this device, a kinematic chain system needs to be solved to obtain a frame transformation from the platform frame $o_p x_p y_p z_p$ to the tether endpoint frame $o_t x_t y_t z_t$. We followed the Denavit–Hartenberg convention [58] to attach reference frames to the kinematic chain, shown in Figure 5, and find the frame transformation. The rotation matrix component of this transformation, R_p^t is given by

$$R_p^t = \begin{bmatrix} -\sin q_2 & -\cos q_2 & 0 \\ -\cos q_2 \sin q_1 & \sin q_1 \sin q_2 & -\cos q_1 \\ \cos q_1 \cos q_2 & -\cos q_1 \sin q_2 & -\sin q_1 \end{bmatrix}, \quad (1)$$

where q_1 and q_2 are the joint angles measured using the potentiometers. Subsequently, the tether azimuth α_p and elevation φ_p angles at the platform endpoint can be obtained from the components of the tether vector $t = \begin{bmatrix} t_x & t_y & t_z \end{bmatrix}^T = R_p^t \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^T$ as

$$\begin{cases} \alpha_p &= \tan^{-1}\left(\frac{t_y}{t_x}\right), \\ \varphi_p &= \left|\tan^{-1}\left(\frac{t_z}{\sqrt{t_x^2 + t_y^2}}\right)\right|. \end{cases} \tag{2}$$

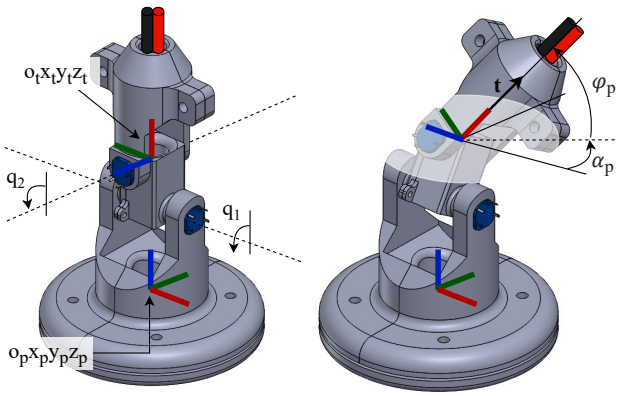


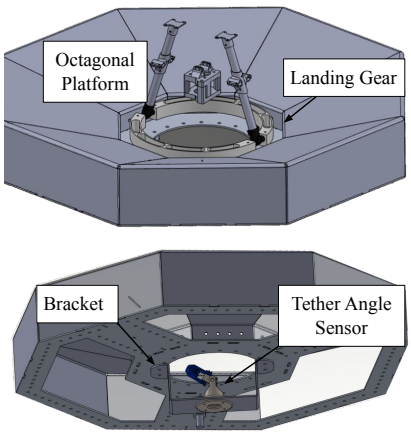
Figure 5. Tether sensor coordinate frames.

3.4. Self-Leveling Landing Platform System

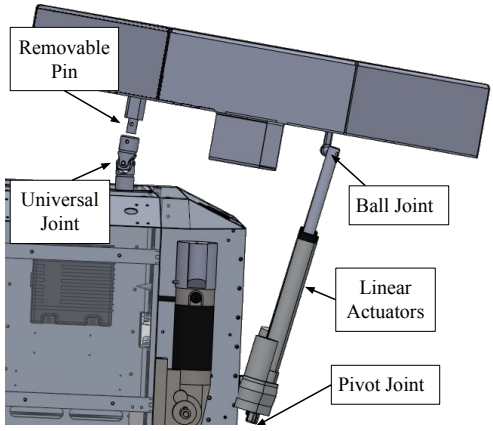
A self-leveling platform was included in the UGV to provide a stable, level surface for the UAV to land on. In our design, we used a tri-axial accelerometer and two linear actuators to adjust the platform attitude and maintain it horizontally leveled, even when the UGV is located on uneven terrain or a slope, as shown in Figure 6a. This platform helps to ensure that the UAV lands safely and precisely without tipping over or damaging itself or its surroundings.



(**a**) Self-leveling platform.



(**b**) Platform parts and landing gear.

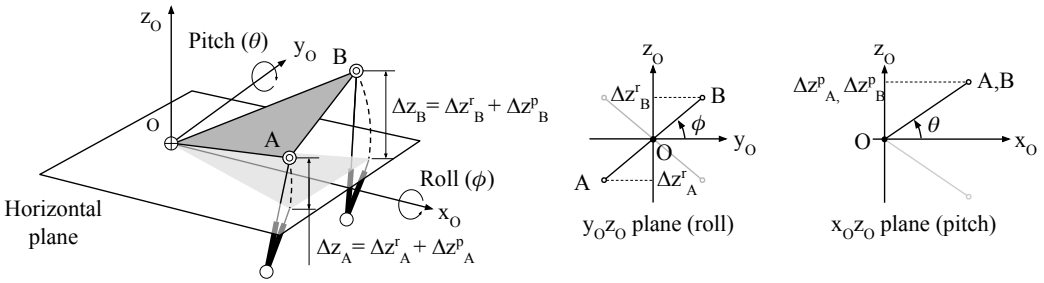


(**c**) Platform mounting.

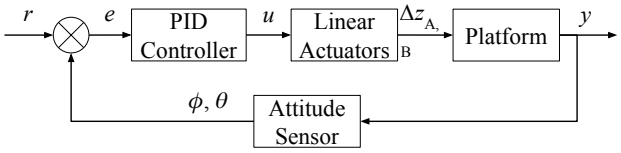
Figure 6. Self-leveling landing platform system: (**a**) the platform compensates for the angles of the UGV by increasing the strokes of the linear actuators; (**b**) the shape of the platform viewed from above and below and the shape of the landing gear that fits in a depression located in the center of the platform; (**c**) side view of the mounting points of the platform.

The landing platform was designed with an octagonal shape and a central depression to serve as a passive positioning mechanism and to securely hold the UAV in place when the UGV moves, as illustrated in Figure 6b. On the UAV side, a landing gear was designed to match the depression in the landing platform. Its circular shape allows the UAV to land without any orientation constraints. This design enables the platform to function effectively as a landing and docking station for the UAV. A bracket to hold the tether angle sensor was mounted below the platform in the center of the depression.

The octagonal platform is connected to three points that are physically attached to two Progressive Automations' PA-14P 12-inch stroke linear actuators and a universal joint, as illustrated in Figure 6c. As expected, the point representing the fixation to the universal joint, O in Figure 7a, does not translate and only rotates. The two points representing the fixation to the linear actuators, A and B, can translate and rotate when the linear actuators increase or decrease their stroke. The displacement of these two points allows for two possible rotation angles of the platform. If A and B move in opposite directions with respect to the horizontal plane, the deck rolls around its x-axis. If A and B move in the same direction, the platform pitches around its y-axis. Therefore, with a combination of these two movements, it is possible to adjust the roll and pitch angles of the platform. A schematic of this movement is shown in Figure 7a.



(**a**) Schematics.



(**b**) Control architecture.

Figure 7. Self-leveling platform control system: (**a**) schematic representation of the platform and its three mounting points, along with illustrations of the roll and pitch actions that can be created using the linear actuators; (**b**) control loop used to level the platform. A PID controller tries to control the roll (ϕ_p) and pitch (θ_p) angles of the platform using a zero attitude reference signal r.

To automatically control the platform's attitude, a closed-loop control system (Figure 7b) was designed using zero roll and pitch angles as setpoints. The controller chosen was the discrete PID control due to its ease of implementation and tuning process. Assuming that the UGV is static and that the only acceleration applied to the system is gravity, the platform attitude is estimated by finding the Euler angles in the transformation

$$\mathbf{a}_b = R_n^b \mathbf{g}_n \,,$$

where $\mathbf{a}_b$ is the gravity acceleration decomposed in the accelerometer's frame b, which in turn is aligned with the platform; R_n^b is the rotation matrix between the inertial and the body frames as a function of the platform attitude angles roll (ϕ_p) and pitch (θ_p); and $\mathbf{g}$ is

the gravity vector in the inertial reference frame n pointing downwards. By rewriting this equation in matrix form, we have:

$$\begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & -\sin\phi_p \\ r_{21} & r_{22} & \sin\phi_p \cos\theta_p \\ r_{31} & r_{32} & \cos\phi_p \cos\theta_p \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}, \tag{3}$$

where g is the module of the acceleration vector. Solving Equation (3) for ϕ and θ, we obtain

$$\begin{cases} \phi_p &= \arctan\left(\frac{a_y}{a_z}\right), \\ \theta_p &= \arcsin\left(-\frac{a_x}{g}\right). \end{cases} \tag{4}$$

which are then used to decompose the gravity acceleration from a tri-axial accelerometer (ADXL345) and used to close the controller's feedback loop.

4. Software Architecture

This section describes the framework developed to operate the TUAV in a stone mine environment. This section is divided into three parts. The first describes the general software architecture, the second describes the mapping procedure, and the third describes the landing procedure.

4.1. Autonomy Framework

For the development of the autonomy framework of the TUAV, we developed a set of ROS nodes to control the drone operation. These nodes were distributed between the computers of the ground and aerial robots. An overview of the UAV and UGV systems and their software is shown in Figure 8. The state machine node selects which mission the TUAV will execute by enabling a mission node. This is done by broadcasting the active node through an ROS topic. The number of mission nodes is variable, depending on the application. The mission nodes generate commands that are sent to the commander node, the only node able to communicate with the actual TUAV. This makes this stack generic and independent of the TUAV platform. In our experiments, for example, we used a DJI drone by simply changing the command node, which is the only platform-dependent node. For our specific setup, the TUAV commander node is able to control the TUAV using auto-takeoff, auto-landing, target position, and target velocity commands. Other than takeoff, which is usually a native function of the drone's controller, in our application, we split the actions of the TUAV into two missions: mapping and landing. The mapping mission node is responsible for moving towards the pillar and executing a lawnmower pattern to map the pillar surface. The landing mission node is responsible for returning the TUAV to the ground vehicle and landing using only tether information. These two missions will be detailed later in this section.

4.2. Velocity Estimation and Localization

Our framework relies on velocity-based control and thus requires that the drone's velocity is reliably estimated. For that, the Intel® RealSense™ T265 odometry solution was integrated directly into the flight controller localization solution, which estimates the drone position and velocity.

Although not explored in this paper, we also developed our localization approach, which uses the tether to obtain relative localization with respect to the landing platform. The complete methodology is explained in [38]. The approach considers a flexible tether with a catenary shape, which can be mathematically modeled. The model, which is based on tether variables such as tether length, tension, and tether angles on both endpoints, can provide a relative localization solution. This solution is fused with the IMU measurements using a sensor fusion algorithm based on the extended Kalman filter (EKF).

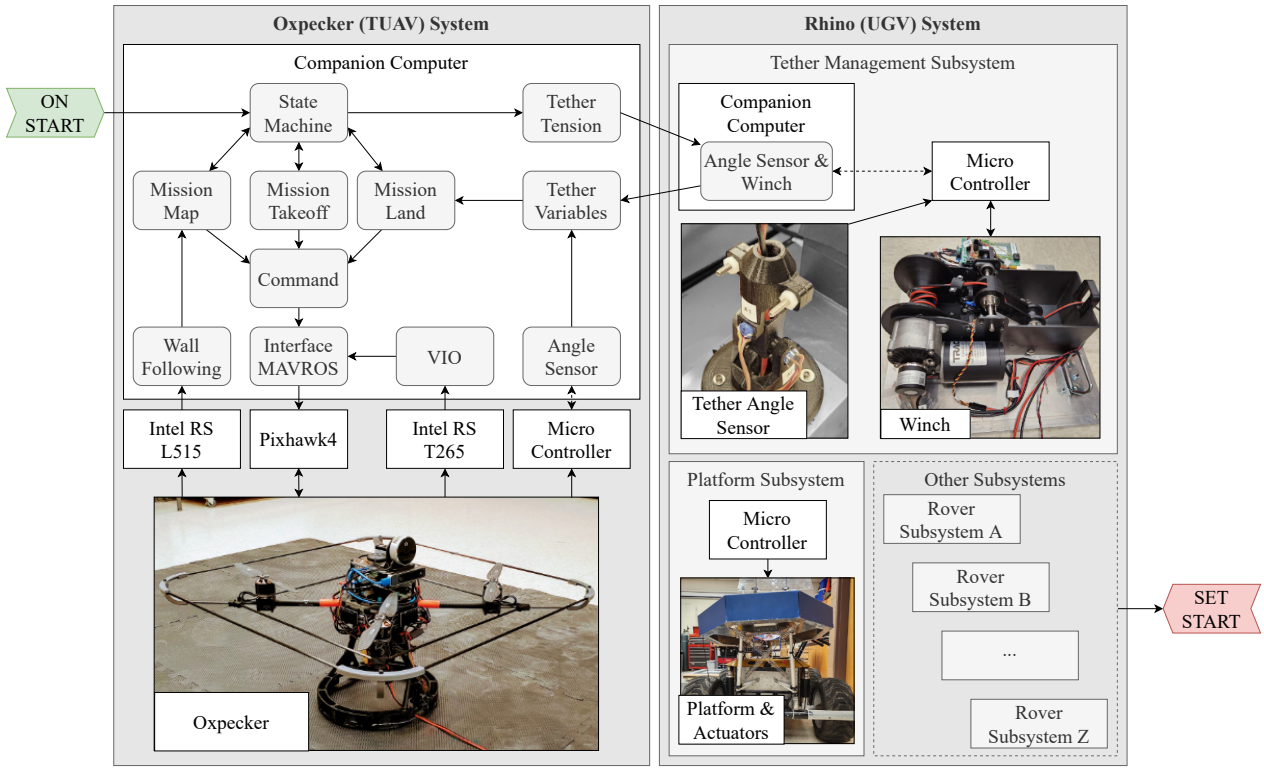


Figure 8. Cooperative autonomous system framework (UAV-UGV). On the **left**: a schematic overview of Oxpecker's system. On the **right**: a schematic overview of Rhino's subsystems related to the drone operation. The rounded gray rectangles represent the ROS nodes developed to operate the autonomous system.

4.3. Wall Coverage

The coverage software on Oxpecker is designed to survey and map the face of a pillar or wall by executing a predefined trajectory. The area to be covered is assumed to be rectangular, and the chosen trajectory is a sweeping pattern, as shown in Figure 9. The sweeping pattern is implemented to cover the entirety of the area, which is done by designing the coverage rows to be perpendicular to the shortest height of the coverage area. As stated in [59], this ensures that the trajectory involves flying the least number of turns, which directly affects the duration, route length, and energy consumption.

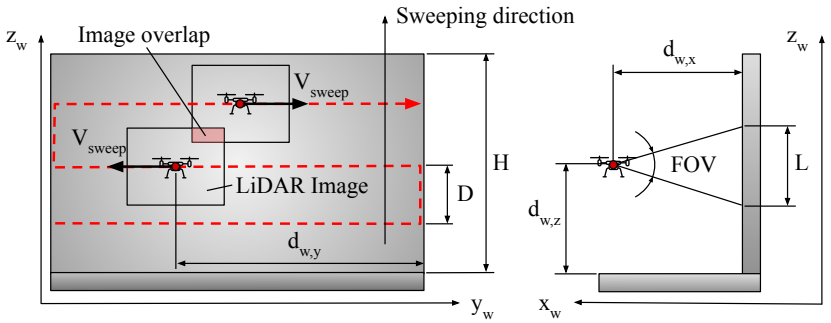


Figure 9. Sweep pattern implemented to cover an entire area with a distance of *D* between coverage rows.

To determine the number of rows required in the pattern, the approach outlined in [60] is followed as was done in our previous work [61]. The first step is to determine the

camera's footprint, L, which can be calculated using the geometric relationship between the distance from the camera plane to the wall, $d_{w,x}$, and the camera's field of view (FOV) [62]:

$$L = 2d_{w,x} \times \tan\left(\frac{\text{FOV}}{2}\right). \tag{5}$$

The footprint is then used to determine the number of coverage rows and the gap needed to acquire a wanted overlap as represented in Figure 9. The number of coverage rows is computed as

$$N = \frac{H}{L(1-s)}, \tag{6}$$

where H is the height of the coverage area, and s is the desired overlap between two images as a fraction. The distance between the coverage rows can be calculated as

$$D = \frac{H}{N}. \tag{7}$$

For the complete design of the trajectory, lateral overlaps must also be considered. Whereas longitudinal overlaps are determined by coverage rows, lateral overlaps are related to the speed of the UAV, V_{sweep}, and the frame rate of the camera as

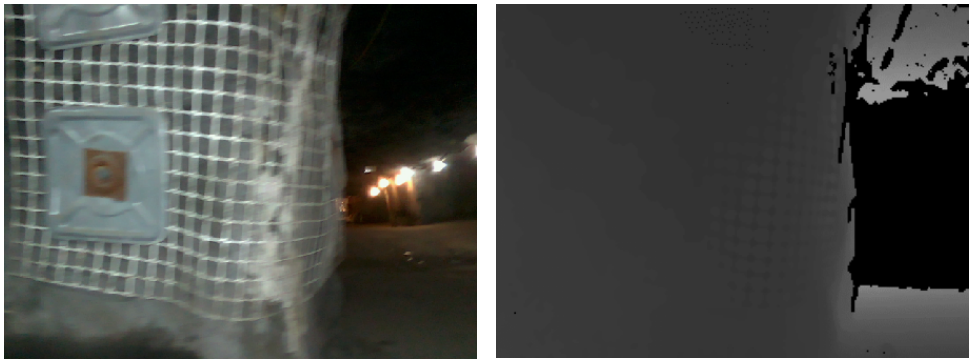
$$V_{\text{sweep}} = \frac{1 - s_{\text{lateral}}}{L \times f}, \tag{8}$$

where s_{lateral} is the overlap expressed as a fraction, L is the image footprint, and f is the frame rate in frames/second.

After defining the survey parameters, the path of the UAV along the coverage rows can be defined by a set of 3D points $(d_{w,x}, d_{w,y}, d_{w,z})$, where $d_{w,x}$ represents the distance from the drone to the wall, $d_{w,y}$ is the distance from the right border of the wall, and $d_{w,z}$ represents the height of a coverage line from the ground. Since we have several rows, the z coordinate is calculated for each row as

$$d_{w,x}^{(i)} = i \times D - \frac{D}{2}, \quad i = 1, \ldots, N_l. \tag{9}$$

Our previous research [61] used visual-inertial odometry (VIO) and predefined waypoints (a sequence of (x, y, z) coordinates) to navigate the drone to follow the desired pattern. However, this method is not practical when mapping walls of different lengths (y coordinates of the path would change for each wall) and was prone to position drift due to the featureless environment. Therefore, in this work, we proposed a reactive method of coverage that relies on flying the drone in front of the wall until it finds its border. An edge-detection software was then developed to replace the necessity of predefined waypoints along the y axis. This software processes the raw depth image given by the drone's LiDAR (RealSense™ L515) and calculates the difference between the average depth of the outer left and outer right sides of the image. If the difference exceeds a set threshold, an edge is detected. Although we chose to use a low-noise sensor, to cope with eventual noise on the depth image given by the sensor, several steps are taken by the system. First, the sensor is configured to work in a given range, thus outputting a constant value for distances outside this range. This can be observed in Figure 10b, which shows a flat black region representing points very far from the sensor. Second, after the depth image is received, the software applies to the image a binary mask that removes the ceiling and the floor from the data, keeping only the region of interest situated in between these two regions. Finally, a sequence of morphological operations (erosions and dilations) are applied to the masked data to remove additional noise.



(**a**) RGB image. (**b**) Depth image.

Figure 10. Raw snapshots from an instance when the edge is detected: (**a**) shows the color image of a pillar in a coal mine (the image is blurry due to the motion of the drone); (**b**) shows a depth image of the same instant. Notice that when the edge is detected, if the drone is leveled, a void appears in the center-right part of the depth image, indicating that the distance in that region is outside of the predefined processing distance. The top-right and bottom-right are ignored, as they can show the ground or ceiling depending on the pillar's height and the drone's relative position.

The x coordinate of the path represents the drone's distance to the wall. In our approach, we keep a fixed distance from the wall using software that also relies on LiDAR data. In addition to keeping a constant distance, the controller adjusts the drone's orientation so that the camera is parallel to the surface being scanned. As illustrated in Figure 11, being parallel and at a fixed distance from the wall is important, especially when the wall is not planar. Furthermore, different distances and orientations can cause consecutive images to not have the appropriate overlap.

To execute the complete path shown in Figure 9, a state machine within the "Mission Map" ROS node (Figure 8) was developed. This node uses the information obtained from the wall-following software (another ROS node) and altimeter to execute the algorithm shown in Figure 12. While the wall-following node maintains a fixed distance from the wall and detects the edges, the altimeter attempts to maintain a specific row height until the next row must be executed.

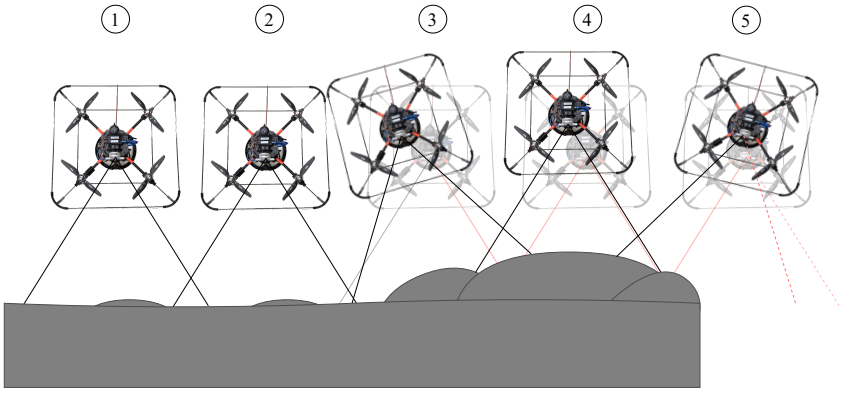


Figure 11. Top view of a UAV scanning the wall (in gray) at five different timestamps. At timestamps 1 and 2, the wall is basically flat, and the drone is able to move safely at a determined distance from the wall. At timestamps 3 and 4, the drone faces a bump in the wall. If the drone keeps moving laterally without adjusting the distance (transparencies), we may have an inconsistent overlap between the images captured at the current and next timestamp. The depictions without transparency illustrate how our controller handles the problem following its contour by adjusting the orientation and distance from the wall. At timestamp 5, the drone is not able to detect a valid distance on the leftmost beam (dashed lines), which indicates an edge of the pillar (see Figure 10) and a need to switch the direction of movement.

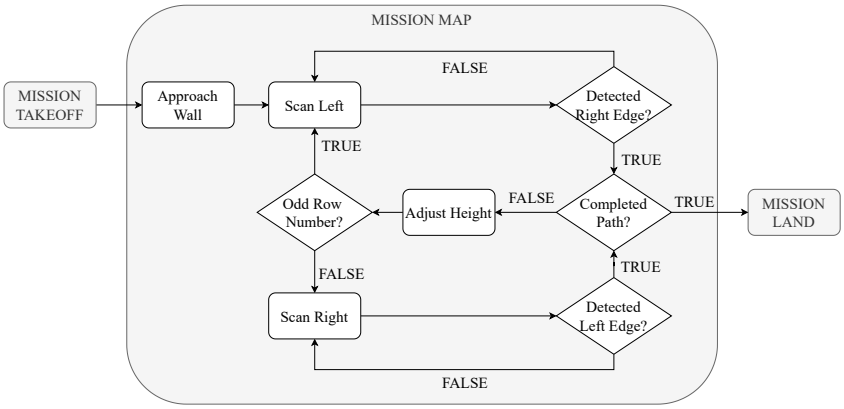


Figure 12. Sweeping trajectory execution algorithm within the "Mission Map" block.

4.4. Tether-Guided Landing

For the TUAV to land accurately on the landing platform situated on the UGV, an algorithm based solely on an altimeter (laser range finder onboard the vehicle) and the tether angle sensors on the TUAV (see Figure 3b) was developed. Therefore, our method does not require a precise localization of the UAV to perform a precise landing on the platform. The ROS node implemented to land the drone is represented by the "Mission Land" block in Figure 8. When this block is activated, the "State Machine" block communicates with the tethering system to set the tension setpoint to a higher value so that the tether tends to become taut, which is the ideal situation for landings. The landing procedure is illustrated in Figure 13. A switched vector field, which represents velocity setpoints given to the low-level controller in the body frame, is used to guide the drone to the landing pad. The transitions in the vector field are obtained from a combination of elevation angle and height, as shown in the flowchart of Figure 13. The direction and magnitude of the velocity command for each action are shown in Table 1.

To summarize the procedure, if the drone is laterally distant enough from the platform, as indicated by a small elevation angle, the flight path obeys the sequence: a 45-degree incline ascent, followed by a horizontal approach, which progresses to a 45-degree descent, ending in a vertical descent. All horizontal movement is directed toward the landing platform. This flight path may shorten, lengthen, or skip one of these actions entirely, depending on how close the TUAV is to the platform, both vertically and horizontally. The transition elevation angles (φ_1, φ_2, φ_3) and heights (h_1, h_2), as well as the command speeds (V and V_d) are kept as parameters for tuning. Another observation is that, although the proposed solution generates a discontinuous vector field, to avoid steps in the desired velocity given as a setpoint to the drone's velocity controller, we linearly change the setpoint when we switch from one phase to the next.

Table 1. Landing actions.

Action	v_x [m s^{-1}]	v_y [m s^{-1}]	v_z [m s^{-1}]
45° Ascent	$-V\cos\alpha\sin\frac{\pi}{4}$	$-V\sin\alpha\sin\frac{\pi}{4}$	$V\cos\frac{\pi}{4}$
XY Approach	$-V\cos\alpha$	$-V\sin\alpha$	0
45° Descent	$-V\cos\alpha\sin\frac{\pi}{4}$	$-V\sin\alpha\sin\frac{\pi}{4}$	$-V\cos\frac{\pi}{4}$
Vertical Descent	$-V_d\cos\alpha$	$-V_d\sin\alpha$	$-V$
Descent w/ P Control	$-V_d\cos\alpha$	$-V_d\sin\alpha$	$-V\frac{h}{h_1}$
Landed/Disarm	0	0	0

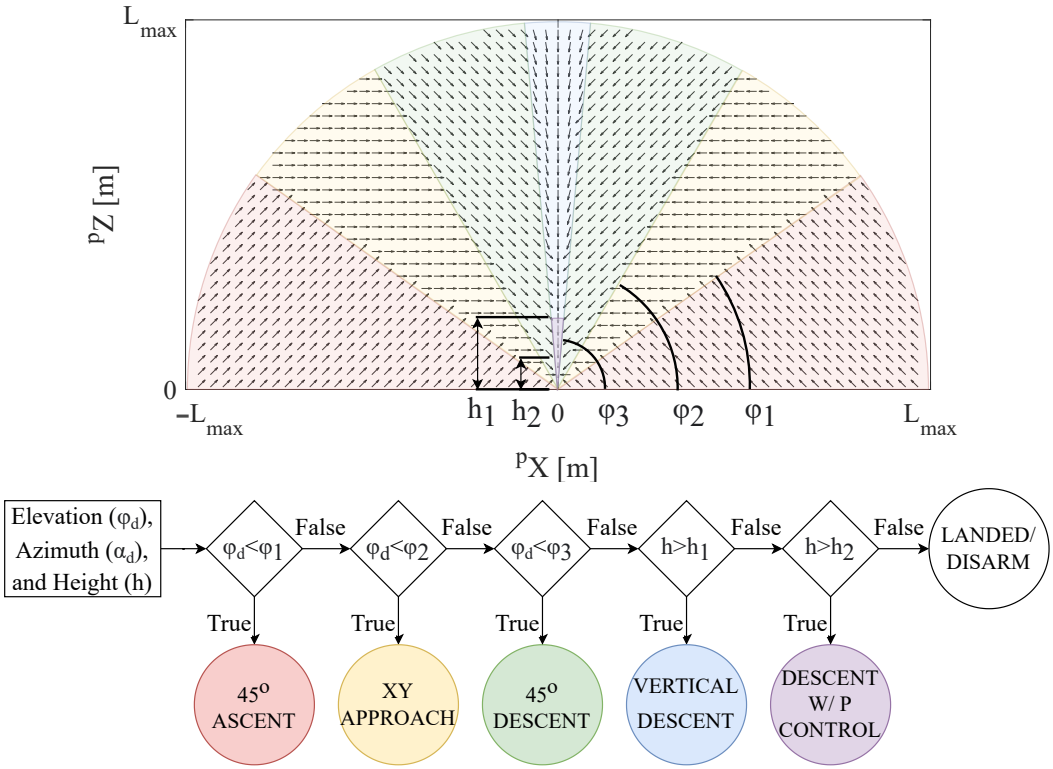


Figure 13. Proposed tether-guided landing approach. The vector field on top shows the direction of movement on the XZ plane (in the platform's coordinate frame), limited by the maximum tether length, L_{max}. The complete 3D vector field can be obtained by rotating this field around the z-axis. As shown in the flowchart, the vector field, although represented in the workspace, is constructed only based on the drone elevation (φ_d) and azimuth (α_d) angles, and the height of the vehicle (h). Therefore, no localization is necessary. Ideally, the drone follows the actions sequentially until landed: 45° ascent, horizontal approach, 45° descent, vertical descent, and vertical descent with P control. The angles φ_1, φ_2, and φ_3, and the heights h_1 and h_2 are thresholds that define the transition between different actions.

5. Field Experiments and Results

This section presents a series of drone experiments conducted in laboratories and mining facilities. These experiments were designed to illustrate and evaluate our system, including mapping and landing approaches. Videos of some experiments can be found at https://www.tinyurl.com/oxpeckeruav (accessed on 12 January 2023).

5.1. UAV Path Planning and Mapping

The path planning strategy described in Section 4.3 was implemented in ROS and tested in a simulated scenario and on a DJI Matrice 100 drone. The latter served as a software integration and testing proxy while the experimental drone hardware was being integrated. Due to the inability to provide a source of odometry ground truth for the physical experiments, our path-planning and mapping approaches were initially tested in a graphical simulator for validation purposes. In one set of simulations, for example, we investigated the impact of combining ranging information with INS and visual odometry [55]. Results from these investigations are summarized in Table 2, where it is evident fusing more information to visual odometry yields improvements in performance.

Table 2. Error associated with different sources of odometry by implementing the path planning strategy (simulator results).

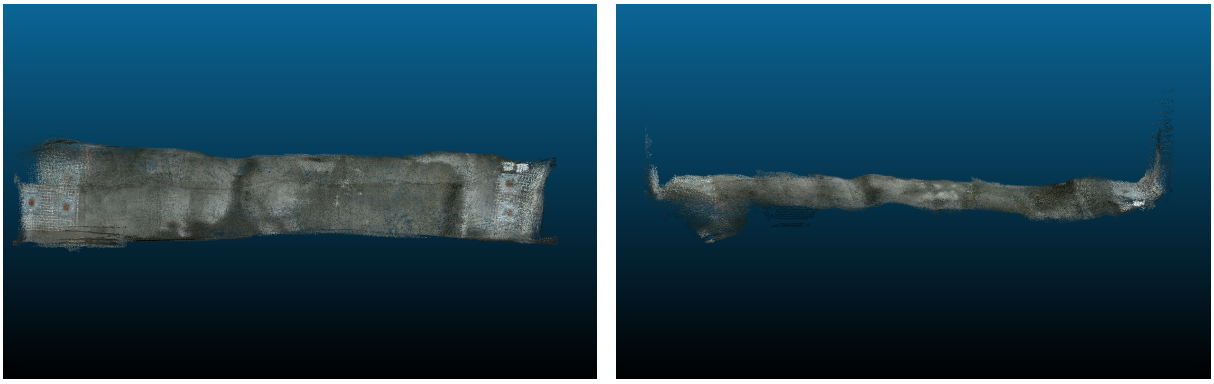
	Max Error (m)			Mean Error (m)			RMSE (m)		
	X	Y	Z	X	Y	Z	X	Y	Z
Visual Odom	0.558	0.779	0.188	0.177	0.405	0.095	0.074	0.180	0.072
Sensor Fusion Odom	0.178	0.162	0.071	0.035	0.067	0.050	0.032	0.067	0.050

Real-world mapping experiments were executed in the Safety Research Coal Mine operated by the National Institute for Occupational Safety and Health (NIOSH), located in Pittsburgh, PA, in the U.S. A wall within the coal mine was surveyed autonomously whilst executing the trajectory described in Section 4.3 and specified by the parameters shown in Table 3. During the flight, the drone's computer saved data that were relevant for mapping. These included RGB images, depth images, and odometry data, which altogether generated approximately 18 MB s^{-1} (i.e., our 2 TB memory card could keep up to 30 h of mapping data). In an effort to reduce the onboard computing cost, the 3D map was created during post-processing using an RGB-D SLAM approach called real-time appearance-based mapping (RTAB-Map) [63]. During post-processing, the primary front-end odometry source was the T265, whereas RTAB-Map used the RGB and depth images to detect loop closures in the back end. The challenge of detecting accurate loop closures in subterranean environments was investigated in [61]. This information was combined with specific parameters for RTAB-Map, which included extracting up to 500 visual features per image and detecting a minimum of 20 visual features. This constraint ensures the acceptance of only certain loop closures. A resulting map can be seen in Figure 14.

By observing the resulting map, it can be seen that the designed trajectory successfully explored and mapped the entire wall. Figure 14a shows that there were no noticeable gaps in the point cloud, indicating sufficient overlap. Figure 14b shows that the left and right edges were found. A video with some other experiments at the NIOSH Coal Mine is available at https://www.tinyurl.com/oxpeckeruav (accessed on 12 January 2023). Additional experiments and simulations that evaluate our coverage methodology can be found in [61].

Table 3. Parameters used to create the lawnmower path and control the drone's movement.

Parameter	Value
Speed	0.35 m/s
Rows	2
Dist. from Wall	1 m, 1.5 m



(**a**) Front view. (**b**) Top view.

Figure 14. Typical map of a pillar. The drone collects the data by moving in a lawnmower pattern in front of the pillar. When the edges of the wall are detected, the drone changes the height and horizontal direction of movement.

5.2. Landing

A series of experiments were performed to demonstrate the method described in Section 4.4 and the tether management system. In these experiments, the TUAV started from different relative positions with respect to the platform. The experiments were conducted in a high bay facility. Figure 15 shows images from the experimental setup. Figure 15a shows Oxpecker hovering above the UGV, with the tether fully extended and taut. In Figure 15b, a sequence of overlaid snapshots shows the drone approaching and landing on the platform on top of Rhino.

For the experiments shown in this section, the transition elevation angles were set to $\varphi_1 = 65°$, $\varphi_2 = 75°$, $\varphi_3 = 85°$ and transition heights to $h_1 = 0.5\,\mathrm{m}$, $h_2 = 0.2\,\mathrm{m}$. The command speed was set to $V = 0.15\,\mathrm{m\,s^{-1}}$. Speed V_d was set to vary with the elevation angle as

$$V_d = \frac{\frac{\pi}{2} - \varphi}{\frac{\pi}{2}}. \tag{10}$$

Notice that this velocity is small for elevation angles close to 90° and is zero when the elevation angle is exactly 90°.



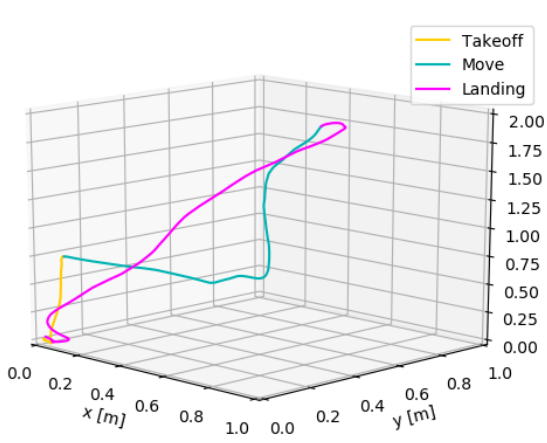
(**a**) Taut tether. (**b**) Landing sequence.

Figure 15. Typical tether-guided landing. The tether is made taut to help guide the drone back to the landing platform carried by the UGV.

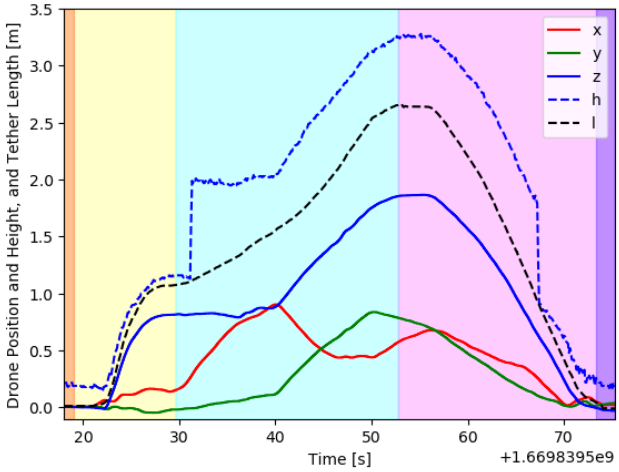
In Figures 16 and 17, we show an overview of one of the experiments conducted to demonstrate the landing methodology. In Figure 16a, the drone executes three missions. In "Mission Takeoff" (yellow trajectory), the drone takes off to a height of 0.8 m from the platform. In "Mission Move" (cyan trajectory), the drone moves forward with a setpoint speed of $0.15\,\mathrm{m\,s^{-1}}$ for 10 s and laterally and up for another 10 s. In "Mission Land" (magenta trajectory), the drone receives zero speed setpoints for 2 s before starting the landing procedure. In Figure 16b, we show the drone localization as given by the T265 sensor, the drone height obtained with the altimeter, and the tether length measured by our system. The background colors represent the different missions described before. Notice that when the altimeter is used for localization, it introduces discontinuities in the data. In this example, the discontinuities appear when the drone leaves the platform, and the altimeter starts measuring the distance to the ground and before landing when the reverse sequence happens.

In Figure 16c, we show the tether tension during the entire experiment. During the first two missions (yellow and cyan backgrounds), the setpoint for the tether tension is 0.3 N. During takeoff, although the controller is trying to release the cable faster to decrease the tension, the drone moves up too fast, and a large peak in the tension occurs. Once the drone starts hovering (around $t = 30\,\mathrm{s}$), the controller is able to adjust to the tension setpoint. Conversely, during landing (magenta background), the drone moves faster than the controller is able to retract the cable, causing the tension to drop below the tension setpoint, which has been increased to help the landing. The landing methodology

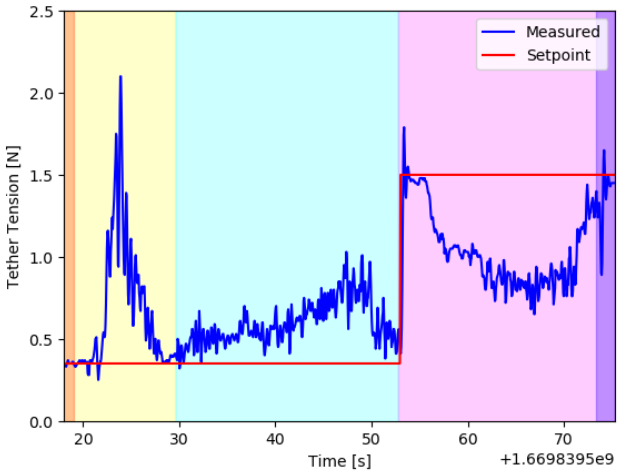
was tested successfully from 10 different starting positions. The results in Figure 16d show the trajectories of the individual flights using the localization solution from the T265 tracking camera.



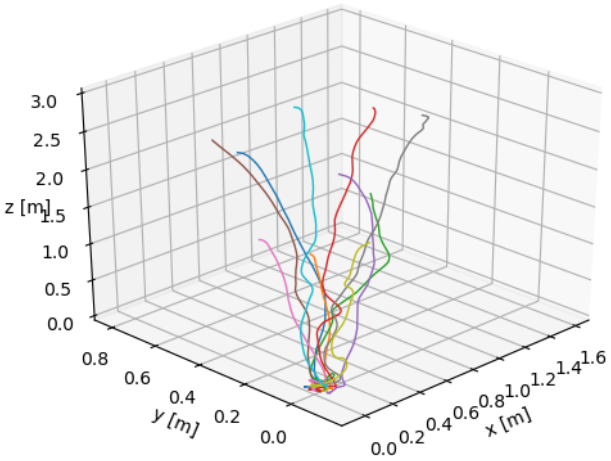
(**a**) Drone trajectory.



(**b**) Drone position and tether length.



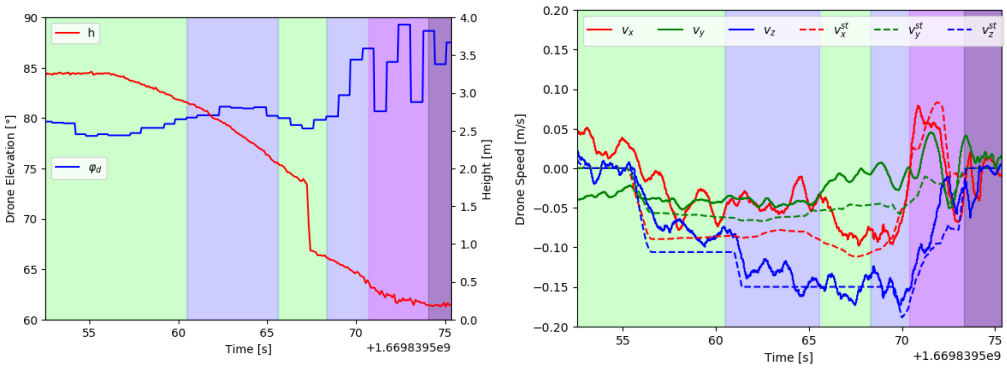
(**c**) Tether tension.



(**d**) Drone trajectories.

Figure 16. Landing experiment: (**a**) TUAV trajectory in a typical experiment; (**b**) drone localization given by the T265 sensor, tether length measured by our system, and height given by a laser altimeter during the landing shown in (**a**); (**c**) tether tension during the landing shown in (**a**); (**d**) trajectories of several landings as seen by the T265 sensor.

Figure 17 shows the variables that control state transitions and the velocity setpoints for different actions during a typical landing. The background colors in this figure match the landing procedure colors in Figure 13. Notice that Oxpecker switches between "45° descent" (green) and "vertical descent" (blue) until it gets very close to the landing spot, as shown by the discontinuity in the measured height in Figure 17a. Once the altimeter measures a height smaller than 0.5 m, it switches to the "descent w/P control" mode until it reaches a height that triggers the "landed" command, which sends zero velocity and disarms the drone.



(**a**) Elevation angle and height.

(**b**) Tether variables.

Figure 17. Variables and drone inputs for the landing shown in Figure 16a. (**a**) Variables that control the state transitions in Figure 13; (**b**) shows velocity commands sent to the drone and measured velocity using the T265 sensor.

6. Conclusions

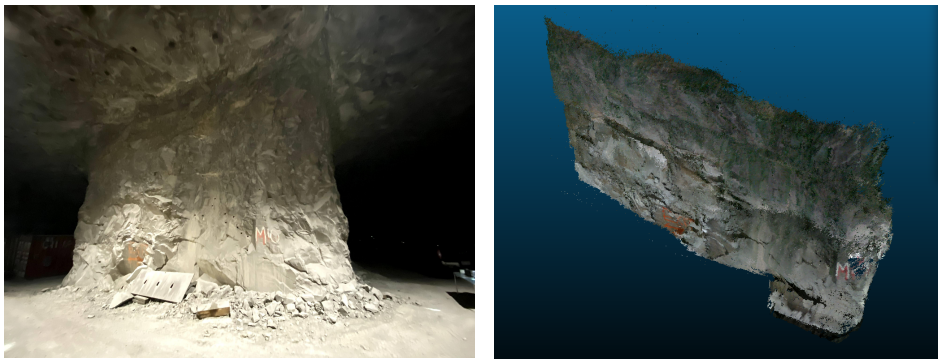
This paper presented the hardware and the software of a tether-powered UAV for underground mining applications. This drone, called Oxpecker, was designed to work in tandem with a UGV, which is responsible for hosting the batteries of the UAV, a landing platform specially designed to allow for precise landing and safe UAV transportation, and a tethering system that controls the tension of the tether and measures its azimuth and elevation angles. The drone itself was constructed around higher voltage (42 V) motors to avoid using voltage converters both on the UAV and on the UGV. Although this solution was successful, we noticed an excessive power loss on the cable, which could be mitigated by transmitting power at even higher voltages through the tether.

Regarding autonomy, the TUAV software relies on the low-level velocity controllers on its Pixhawk controller, which is connected to the Intel® RealSense™ T265 tracker sensor for velocity estimation. Thus, our software for stone mine pillar surveys and precise landing, described and demonstrated in the paper, sends velocity commands to the drone. In contrast to previous work, for both tasks, instead of localizing the UAV using vision or even the tether, which we also showed to be possible in our previous work [38], we developed reactive approaches that only depend on instantaneous measurements given by the sensors. For pillar surveys, the vehicle, based on LiDAR data, is controlled to follow the wall until the wall edges are encountered. At this moment, the drone's height is adjusted (based on an altimeter) to create a lawnmower pattern in front of the wall. In the same spirit, for landing, the angles of the tether measured on the vehicle are used to create a vector field that precisely guides the drone to its landing platform. Experiments that illustrate and validate our strategies are presented in the paper.

This paper also describes a tether sensing and management system constructed based on an electric winch and several sensors that measure the length of cable deployed, the azimuth and elevation angles of the tether, and the tether tension. Given tension measurements, a microcontroller is able to actuate the winch to adjust the tension to values that will help the TUAV perform its tasks while simultaneously avoiding tangling and contact between the tether and the ground. Although tether length and angles were not used by the software presented in this paper, they are essential for tether-based localization [38].

Future work will include a complete and long-term evaluation of the system in actual limestone mines. Although we have initial results in such mines (see the map of a pillar in Figure 18), we never had the opportunity to test the complete robotic system in such an environment. Regarding the hardware, the drone power system will require our attention. We intend to abandon the idea of avoiding converters and add a DC-DC converter to increase the voltage on the UGV and another to step this down on the TUAV. This will allow for lighter cables, which can eventually compensate for the weight of the converter. Moreover, we expect to reduce the power loss by at least ten times. We are currently

working on developing software that uses a continuous vector field for landing. We expect this method to have fewer parameters to adjust than the current strategy and will also allow the drone to land faster. We have recently implemented a localization system based on the tether, but we are still relying on the T265 camera to provide the information needed by Pixhawk to control the drone. Our future work will then integrate our localization system into Oxpecker's MCU. We believe this will increase the system's robustness, especially in very low-light conditions.



(**a**) Underground stone mine pillar. (**b**) 3D map of the pillar face.

Figure 18. Stone mine pillar and 3D map of one of its faces obtained with a DJI M100 drone.

Author Contributions: Conceptualization, B.M.R.J., R.R.L., K.S., J.R., J.N.G. and G.A.S.P.; methodology, B.M.R.J., R.R.L., K.S., J.R., J.N.G. and G.A.S.P.; software, B.M.R.J., R.R.L., K.S., J.R. and G.A.S.P.; validation, B.M.R.J., R.R.L., K.S., J.R. and G.A.S.P.; formal analysis, B.M.R.J., R.R.L. and G.A.S.P.; investigation, B.M.R.J., R.R.L., K.S., J.R., J.N.G. and G.A.S.P.; resources, J.N.G. and G.A.S.P.; data curation, B.M.R.J., R.R.L., K.S. and J.R.; writing—original draft preparation, B.M.R.J., R.R.L. and G.A.S.P.; writing—review and editing, B.M.R.J., R.R.L., K.S., J.R., J.N.G. and G.A.S.P.; visualization, B.M.R.J., R.R.L. and K.S.; supervision, J.N.G. and G.A.S.P.; project administration, J.N.G. and G.A.S.P.; funding acquisition, J.N.G. and G.A.S.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Alpha Foundation for the Improvement of Mine Safety and Health, Inc. (ALPHA FOUNDATION) grant number AFC820-69. The views, opinions, and recommendations expressed herein are solely those of the authors and do not imply any endorsement by ALPHA FOUNDATION, its directors, and its staff.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to thank our collaborators on the research project, Ihsan "Berk" Tulu, Yu Gu, and several WVU Robotics students. Special thanks go to Dylan Covell for his participation in the design discussions and our collaborators in the stone mine and NIOSH's Safety Research Coal Mine and Experimental Mine. Bernardo Martinez Rocamora Jr. is a Statler College fellow.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

GNSS	Global Navigation Satellite Systems
3D	Three Dimensional
VIO	Visual Inertial Odometry
LiDAR	Light Detection and Ranging
UAV	Unmanned Aerial Vehicle
TUAV	Tethered Unmanned Aerial Vehicle
UGV	Unmanned Ground Vehicle

ROS	Robot Operating System
ESC	Electronic Speed Controller
IMU	Inertial Measurement Unit
GPS	Global Positioning System
GLONASS	Global Navigation Satellite System
SSD	Solid-State Drive
MCU	Microcontroller Unit
EKF	Extended Kalman Filter
SLAM	Simultaneous Localization and Mapping
V-SLAM	Visual Simultaneous Localization and Mapping
PID	Proportional Integral Derivative
ADC	Analog to Digital Converter
FOV	Field of View

Appendix A. List of Components

Table A1 presents the list of commercially available components of Oxpecker.

Table A1. TUAV components—power and weight specifications.

Part Name	Manufacturer	Model	Quantity	Power (W)	Weight (g)
Quadrotor Frame	Holybro	X500	1	-	470
Brushless Motors	T-MOTOR	MN3520 KV400	4	680	888
Propellers	T-MOTOR	P12x4 Prop	4	-	58
Electronic Speed Controllers	T-MOTOR	FLAME HV 60A	4	-	294
Flight Management Unit	Holybro	Pixhawk 4	1	2.5	16
Power Management Board	Holybro	PM07	1	-	40
Telemetry Transmitter	Holybro	TT Radio V3 915 MHz	1	0.5	38
GPS Antenna	Holybro	Pixhawk 4 GPS Module	1	0.8	32
Companion Computer	Intel®	NUC11i5PAH (without case)	1	36	210
LiDAR Camera	Intel®	RealSense™ L515	1	3.3	95
Tracking Camera	Intel®	RealSense™ T265	1	1.5	55
UWB Transceiver Module	Decawave	DWM1001	1	0.3	-
Distance Measurement Sensor	Garmin	LidarLite3	1	0.7	22
Distance Measurement Sensor	Garmin	LidarLite4	1	0.7	22
Inertial Measurement Unit	Analog Devices	ADIS-16495 2BMLZ-ND	1	0.3	42
RC Receiver	Spektrum	AR620 DSMX 6-Channel	1	-	8
Microcontroller	Arduino	UNO R3	1	0.2	25
Flexible Wire (16AWG, 2 cond.)	BNTECHGO	-	15 m	-	570

References

1. Said, K.O.; Onifade, M.; Githiria, J.M.; Abdulsalam, J.; Bodunrin, M.O.; Genc, B.; Johnson, O.; Akande, J.M. On the application of drones: A progress report in mining operations. *Int. J. Min. Reclam. Environ.* **2021**, *35*, 235–267. [CrossRef]
2. Shahmoradi, J.; Talebi, E.; Roghanchi, P.; Hassanalian, M. A comprehensive review of applications of drone technology in the mining industry. *Drones* **2020**, *4*, 34. [CrossRef]
3. Tong, X.; Liu, X.; Chen, P.; Liu, S.; Luan, K.; Li, L.; Liu, S.; Liu, X.; Xie, H.; Jin, Y.; et al. Integration of UAV-based photogrammetry and terrestrial laser scanning for the three-dimensional mapping and monitoring of open-pit mine areas. *Remote Sens.* **2015**, *7*, 6635–6662. [CrossRef]
4. Kim, D.P.; Kim, S.B.; Back, K.S. Analysis of Mine Change Using 3D Spatial Information Based on Drone Image. *Sustainability* **2022**, *14*, 3433. [CrossRef]
5. Ge, L.; Li, X.; Ng, A.H.M. UAV for mining applications: A case study at an open-cut mine and a longwall mine in New South Wales, Australia. In Proceedings of the 2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Beijing, China, 10–15 July 2016; pp. 5422–5425.
6. Battulwar, R.; Winkelmaier, G.; Valencia, J.; Naghadehi, M.Z.; Peik, B.; Abbasi, B.; Parvin, B.; Sattarvand, J. A practical methodology for generating high-resolution 3D models of open-pit slopes using UAVs: Flight path planning and optimization. *Remote Sens.* **2020**, *12*, 2283. [CrossRef]
7. Padró, J.C.; Cardozo, J.; Montero, P.; Ruiz-Carulla, R.; Alcañiz, J.M.; Serra, D.; Carabassa, V. Drone-based identification of erosive processes in open-pit mining restored areas. *Land* **2022**, *11*, 212. [CrossRef]

8. Herrmann, E.; Jackisch, R.; Zimmermann, R.; Gloaguen, R.; Lünich, K.; Kieschnik, L. Drone-borne spectral monitoring of post-mining areas. *Geophys. Res. Abstr.* **2019**, *21*, EGU2019-15001.
9. Turner, R.; Bhagwat, N.; Galayda, L.; Knoll, C.; Russell, E.; MacLaughlin, M. Geotechnical Characterization of Underground Mine Excavations from UAV-Captured Photogrammetric & Thermal Imagery. In Proceedings of the 52nd US Rock Mechanics/Geomechanics Symposium, Seattle, WA, USA, 17–20 June 2018; OnePetro: Richardson, TX, USA, 2018.
10. Russell, E.A.; MacLaughlin, M.; Turner, R. UAV-based geotechnical modeling and mapping of an inaccessible underground site. In Proceedings of the 52nd US Rock Mechanics/Geomechanics Symposium, Seattle, WA, USA, 17–20 June 2018; OnePetro: Richardson, TX, USA, 2018.
11. Dunnington, L.; Nakagawa, M. Fast and safe gas detection from underground coal fire by drone fly over. *Environ. Pollut.* **2017**, *229*, 139–145. [CrossRef]
12. De Petrillo, M.; Beard, J.; Gu, Y.; Gross, J.N. Search planning of a uav/ugv team with localization uncertainty in a subterranean environment. *IEEE Aerosp. Electron. Syst. Mag.* **2021**, *36*, 6–16. [CrossRef]
13. De Croon, G.; De Wagter, C. Challenges of autonomous flight in indoor environments. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 1003–1009.
14. Azpúrua, H.; Saboia, M.; Freitas, G.M.; Clark, L.; Agha-mohammadi, A.a.; Pessin, G.; Campos, M.F.; Macharet, D.G. A Survey on the autonomous exploration of confined subterranean spaces: Perspectives from real-word and industrial robotic deployments. *Robot. Auton. Syst.* **2023**, *160*, 104304. [CrossRef]
15. Chowdhary, G.; Johnson, E.N.; Magree, D.; Wu, A.; Shein, A. GPS-denied indoor and outdoor monocular vision aided navigation and control of unmanned aircraft. *J. Field Robot.* **2013**, *30*, 415–438. [CrossRef]
16. Cvišić, I.; Ćesić, J.; Marković, I.; Petrović, I. SOFT-SLAM: Computationally efficient stereo visual simultaneous localization and mapping for autonomous unmanned aerial vehicles. *J. Field Robot.* **2018**, *35*, 578–595. [CrossRef]
17. Ebadi, K.; Bernreiter, L.; Biggie, H.; Catt, G.; Chang, Y.; Chatterjee, A.; Denniston, C.E.; Deschênes, S.P.; Harlow, K.; Khattak, S.; et al. Present and future of slam in extreme underground environments. *arXiv* **2022**, arXiv:2208.01787.
18. Zhang, J.; Singh, S. Low-drift and real-time lidar odometry and mapping. *Auton. Robot.* **2017**, *41*, 401–416. [CrossRef]
19. Khattak, S.; Mascarich, F.; Dang, T.; Papachristos, C.; Alexis, K. Robust thermal-inertial localization for aerial robots: A case for direct methods. In Proceedings of the 2019 IEEE International Conference on Unmanned Aircraft Systems (ICUAS), Atlanta, GA, USA, 11–14 June 2019; pp. 1061–1068.
20. Quigley, M.; Gerkey, B.; Conley, K.; Faust, J.; Foote, T.; Leibs, J.; Berger, E.; Wheeler, R.; Ng, A. ROS: An open-source Robot Operating System. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA) Workshop on Open Source Robotics, Kobe, Japan, 12–17 May 2009.
21. Neumann, T.; Ferrein, A.; Kallweit, S.; Scholl, I. Towards a mobile mapping robot for underground mines. In Proceedings of the 2014 PRASA, RobMech and AfLaT International Joint Symposium, Cape Town, South Africa, 27–28 November 2014; pp. 27–28.
22. Gill, P.S.; Hatfield, M.C.; Randle, D.; Wies, R.; Ganguli, R.; Rosetti, S.; Vanderwaal, S. Team of Unmanned Aircraft Systems (UAS) and Unmanned Ground Vehicles (UGV) for Emergency Response in Mining Applications. In Proceedings of the 51st AIAA/SAE/ASEE Joint Propulsion Conference, Orlando, FL, USA, 27–29 July 2015; p. 4111.
23. Tranzatto, M.; Miki, T.; Dharmadhikari, M.; Bernreiter, L.; Kulkarni, M.; Mascarich, F.; Andersson, O.; Khattak, S.; Hutter, M.; Siegwart, R.; et al. CERBERUS in the DARPA Subterranean Challenge. *Sci. Robot.* **2022**, *7*, eabp9742. [CrossRef] [PubMed]
24. Hudson, N.; Talbot, F.; Cox, M.; Williams, J.; Hines, T.; Pitt, A.; Wood, B.; Frousheger, D.; Surdo, K.L.; Molnar, T.; et al. Heterogeneous Ground and Air Platforms, Homogeneous Sensing: Team CSIRO Data61's Approach to the DARPA Subterranean Challenge. *arXiv* **2021**, arXiv:2104.09053.
25. Best, G.; Garg, R.; Keller, J.; Hollinger, G.A.; Scherer, S. Resilient Multi-Sensor Exploration of Multifarious Environments with a Team of Aerial Robots. In Proceedings of the Robotics: Science and Systems (RSS), New York, NY, USA, 27 June–1 July 2022.
26. Agha, A.; Otsu, K.; Morrell, B.; Fan, D.D.; Thakker, R.; Santamaria-Navarro, A.; Kim, S.K.; Bouman, A.; Lei, X.; Edlund, J.; et al. Nebula: Quest for robotic autonomy in challenging environments; team costar at the darpa subterranean challenge. *arXiv* **2021**, arXiv:2103.11470.
27. Rouček, T.; Pecka, M.; Čížek, P.; Petříček, T.; Bayer, J.; Šalanskỳ, V.; Azayev, T.; Heřt, D.; Petrlík, M.; Báča, T.; et al. System for multi-robotic exploration of underground environments CTU-CRAS-NORLAB in the DARPA Subterranean Challenge. *arXiv* **2021**, arXiv:2110.05911.
28. Ohradzansky, M.T.; Rush, E.R.; Riley, D.G.; Mills, A.B.; Ahmad, S.; McGuire, S.; Biggie, H.; Harlow, K.; Miles, M.J.; Frew, E.W.; et al. Multi-agent autonomy: Advancements and challenges in subterranean exploration. *arXiv* **2021**, arXiv:2110.04390.
29. De Petris, P.; Nguyen, H.; Dharmadhikari, M.; Kulkarni, M.; Khedekar, N.; Mascarich, F.; Alexis, K. RMF-owl: A collision-tolerant flying robot for autonomous subterranean exploration. *arXiv* **2022**, arXiv:2202.11055.
30. Flyability. Elios 3—Digitizing the Inaccessible. 2022. Available online: https://info.flyability.com/elios-3-inspection-drone (accessed on 12 January 2023).
31. De Petris, P.; Khattak, S.; Dharmadhikari, M.; Waibel, G.; Nguyen, H.; Montenegro, M.; Khedekar, N.; Alexis, K.; Hutter, M. Marsupial Walking-and-Flying Robotic Deployment for Collaborative Exploration of Unknown Environments. *arXiv* **2022**, arXiv:2205.05477.

32. Papachristos, C.; Tzes, A. The power-tethered UAV-UGV team: A collaborative strategy for navigation in partially-mapped environments. In Proceedings of the 22nd IEEE Mediterranean Conference on Control and Automation, Palermo, Italy, 31 May–2 June 2014; pp. 1153–1158.
33. Walendziuk, W.; Oldziej, D.; Slowik, M. Power supply system analysis for tethered drones application. In Proceedings of the 2020 IEEE International Conference Mechatronic Systems and Materials (MSM), Bialystok, Poland, 1–3 July 2020; pp. 1–6.
34. Jain, K.P.; Kotaru, P.; de Sa, M.; Mueller, M.W.; Sreenath, K. Tethered Power Supply for Quadcopters: Architecture, Analysis and Experiments. *arXiv* **2022**, arXiv:2203.08180.
35. Lee, D.; Zhou, J.; Lin, W.T. Autonomous battery swapping system for quadcopter. In Proceedings of the 2015 IEEE International Conference on Unmanned Aircraft Systems (ICUAS), Denver, CO, USA, 9–12 June 2015; pp. 118–124.
36. Alyassi, R.; Khonji, M.; Karapetyan, A.; Chau, S.C.K.; Elbassioni, K.; Tseng, C.M. Autonomous recharging and flight mission planning for battery-operated autonomous drones. *IEEE Trans. Autom. Sci. Eng.* **2022**.. [CrossRef]
37. Boukoberine, M.N.; Zhou, Z.; Benbouzid, M. Power supply architectures for drones-a review. In Proceedings of the IECON 2019-45th Annual Conference of the IEEE Industrial Electronics Society, Lisbon, Portugal, 14–17 October 2019; Volume 1, pp. 5826–5831.
38. Lima, R.R.; Pereira, G.A. On the Development of a Tether-based Drone Localization System. In Proceedings of the 2021 IEEE International Conference on Unmanned Aircraft Systems (ICUAS), Athens, Greece, 15–18 June 2021; pp. 195–201.
39. Al-Radaideh, A.; Sun, L. Observability Analysis and Bayesian Filtering for Self-Localization of a Tethered Multicopter in GPS-Denied Environments. In Proceedings of the 2019 International Conference on Unmanned Aircraft Systems (ICUAS), Atlanta, GA, USA, 11–14 June 2019; pp. 1041–1047.
40. Borgese, A.; Guastella, D.C.; Sutera, G.; Muscato, G. Tether-Based Localization for Cooperative Ground and Aerial Vehicles. *IEEE Robot. Autom. Lett.* **2022**, *7*, 8162–8169. [CrossRef]
41. Alarcón, F.; García, M.; Maza, I.; Viguria, A.; Ollero, A. A Precise and GNSS-Free Landing System on Moving Platforms for Rotary-Wing UAVs. *Sensors* **2019**, *19*, 886. [CrossRef]
42. Al-Radaideh, A.; Sun, L. Self-Localization of Tethered Drones without a Cable Force Sensor in GPS-Denied Environments. *Drones* **2021**, *5*, 135. [CrossRef]
43. Xiao, X.; Fan, Y.; Dufek, J.; Murphy, R. Indoor UAV Localization Using a Tether. In Proceedings of the 2018 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), Philadelphia, PA, USA, 6–8 August 2018; pp. 1–6.
44. Chang, C.W.; Lo, L.Y.; Cheung, H.C.; Feng, Y.; Yang, A.S.; Wen, C.Y.; Zhou, W. Proactive Guidance for Accurate UAV Landing on a Dynamic Platform: A Visual–Inertial Approach. *Sensors* **2022**, *22*, 404. [CrossRef]
45. Wang, J.; Wang, T.; He, Z.; Cai, W.; Sun, C. Towards better generalization in quadrotor landing using deep reinforcement learning. *Appl. Intell.* **2022**, 1–19. . [CrossRef]
46. Keipour, A.; Pereira, G.A.; Bonatti, R.; Garg, R.; Rastogi, P.; Dubey, G.; Scherer, S. Visual Servoing Approach to Autonomous UAV Landing on a Moving Vehicle. *Sensors* **2022**, *22*, 6549. [CrossRef]
47. Gonçalves, V.M.; McLaughlin, R.; Pereira, G.A. Precise landing of autonomous aerial vehicles using vector fields. *IEEE Robot. Autom. Lett.* **2020**, *5*, 4337–4344. [CrossRef]
48. Alarcón, F.; Santamaría, D.; Viguria, A.; Ollero, A.; Heredia, G. Helicopter GNC System for Autonomous Landing by Using a Tether in a GPS Denied Scenario. In Proceedings of the 2015 International Conference on Unmanned Aircraft Systems (ICUAS), Denver, CO, USA, 9–12 June 2015; pp. 1067–1073.
49. Oh, S.R.; Pathak, K.; Agrawal, S.K.; Pota, H.R.; Garratt, M. Approaches for a Tether-guided Landing of an Autonomous Hlicopter. *IEEE Trans. Robot.* **2006**, *22*, 536–544.
50. Mfiri, J.T.; Treurnicht, J.; Engelbrecht, J.A.A. Automated Landing of a Tethered Quad-rotor UAV with Constant Winching Force. In Proceedings of the 2016 Pattern Recognition Association of South Africa and Robotics and Mechatronics International Conference (PRASA-RobMech), Stellenbosch, South Africa, 30 November–2 December 2016; pp. 1–6.
51. Mulgaonkar, Y.; Whitzer, M.; Morgan, B.; Kroninger, C.M.; Harrington, A.M.; Kumar, V. Power and weight considerations in small, agile quadrotors. In Proceedings of the Micro-and Nanotechnology Sensors, Systems, and Applications VI, Baltimore, MD, USA, 5–9 May 2014; SPIE: Bellingham, WA, USA, 2014; Volume 9083, pp. 376–391.
52. Foehn, P.; Kaufmann, E.; Romero, A.; Penicka, R.; Sun, S.; Bauersfeld, L.; Laengle, T.; Cioffi, G.; Song, Y.; Loquercio, A.; et al. Agilicious: Open-source and open-hardware agile quadrotor for vision-based flight. *Sci. Robot.* **2022**, *7*, eabl6259. [CrossRef] [PubMed]
53. Koubâa, A.; Allouch, A.; Alajlan, M.; Javed, Y.; Belghith, A.; Khalgui, M. Micro air vehicle link (mavlink) in a nutshell: A survey. *IEEE Access* **2019**, *7*, 87658–87680. [CrossRef]
54. Ermakov, V. MAVROS. 2020. Available online: https://github.com/mavlink/mavros (accessed on 12 January 2023).
55. Samarakoon, K.Y. UAV Path Planning and Multi-Modal Localization for Mapping in a Subterranean Environment. Master's Thesis, West Virginia University, Morgantown, WV, USA, 2022.
56. Sivaneri, V.O.; Gross, J.N. UGV-to-UAV cooperative ranging for robust navigation in GNSS-challenged environments. *Aerosp. Sci. Technol.* **2017**, *71*, 245–255. [CrossRef]
57. Clarke, J.; Mills, S. Sensor Evaluation for Voxel-Based RGB-D SLAM. In Proceedings of the 2021 36th IEEE International Conference on Image and Vision Computing New Zealand (IVCNZ), Tauranga, New Zealand, 8–10 December 2021; pp. 1–6.

58. Denavit, J.; Hartenberg, R.S. A kinematic notation for lower-pair mechanisms based on matrices. *J. Appl. Mech.* **1955**, *22*, 215–221. [CrossRef]
59. Li, Y.; Chen, H.; Er, M.J.; Wang, X. Coverage path planning for UAVs based on enhanced exact cellular decomposition method. *Mechatronics* **2011**, *21*, 876–885. [CrossRef]
60. Avellar, G.S.; Pereira, G.A.; Pimenta, L.C.; Iscold, P. Multi-UAV routing for area coverage and remote sensing with minimum time. *Sensors* **2015**, *15*, 27783–27803. [CrossRef]
61. Samarakoon, K.Y.; Pereira, G.A.; Gross, J.N. Impact of the Trajectory on the Performance of RGB-D SLAM Executed by a UAV in a Subterranean Environment. In Proceedings of the 2022 IEEE International Conference on Unmanned Aircraft Systems (ICUAS), Dubrovnik, Croatia, 21–24 June 2022; pp. 812–820.
62. He, J.; Li, Y.; Zhang, K. Research of UAV flight planning parameters. *Positioning* **2012**, *4*, 43–45. [CrossRef]
63. Labbé, M.; Michaud, F. RTAB-Map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation. *J. Field Robot.* **2019**, *36*, 416–446. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.