

TDP: Covenant (Mexico), Outdoors Category

Espinosa Perez Marlon, Fuentes Rangel Jesus Eduardo, Hernandez Diaz Rodrigo Sebastian, Ochoa Obezo Manuel
Eduardo & Martinez Virgen Kevin Alan

Covenant

ABSTRACT

This paper presents an autonomous unmanned aerial vehicle (UAV) system designed to address three critical challenges in disaster response scenarios: mapping of disaster areas, person identification and location, and water delivery operations. The system integrates advanced computer vision technologies with the ZED X[1] stereo camera at the core for 3D mapping and object detection, OpenAI's GPT-4 API for intelligent person identification, and a custom servo-controlled water release mechanism. For mapping operations, we utilize the ZED SDK's simultaneous localization and mapping (SLAM)[9] capabilities to generate real-time point cloud data transmitted via a 4G cellular modem. Person detection leverages the Stereo Labs object tracking combined with natural language processing through OpenAI's API to match verbal descriptions with visual identifications. The water delivery system employs a bucket mechanism suspended by a 2-meter rope with a servo motor-controlled release system for precise water deployment.

1 INTRODUCTION

Natural disasters pose significant challenges to emergency response teams, requiring rapid assessment of affected areas, location of survivors, and delivery of essential supplies. Traditional response methods often face limitations due to dangerous conditions, inaccessible terrain, and time constraints that can prove critical in life-saving operations. Unmanned aerial vehicles have emerged as powerful tools for disaster response, offering unique capabilities for surveillance, search and rescue, and supply delivery in environments too hazardous or difficult for human responders to access safely.

The International Micro Air Vehicle (IMAV) competition challenges teams to develop innovative solutions that address real-world scenarios through autonomous flight systems. This paper presents our multi-challenge approach that integrates three distinct mission capabilities into a single autonomous platform: comprehensive disaster area mapping, intelligent person identification and location, and targeted water delivery operations.

Recent advances in computer vision, artificial intelligence, and embedded systems have made it possible to deploy sophisticated autonomous systems on relatively small UAV platforms. The ZED X stereo camera represents a significant advancement in robotics perception, offering human-like depth perception from 10 cm to tens of meters with advanced AI capabilities. OpenAI's vision fine-tuning capabilities have demonstrated remarkable improvements in object detection and visual understanding tasks, making intelligent person identification feasible for autonomous systems.

Our integrated approach addresses the critical need for versatile emergency response UAVs capable of performing multiple mission types without requiring separate specialized platforms. This paper details the technical implementation of each challenge solution.

2 DISASTER AREA MAPPING SOLUTION

2.1 Hardware Configuration

The mapping solution utilizes the ZED X stereo camera from Stereolabs, specifically designed for outdoor robotics applications. The ZED X uses advanced neural networks[10], providing advanced stereo depth sensing technology with high precision 3D positioning capabilities.

The ZED X camera offers several advantages for disaster mapping applications. Built to withstand all weather conditions, the ZED X is ideal for autonomous robots in challenging outdoor environments, particularly suited for obstacle detection and navigation. The camera provides real-time depth maps at various resolutions and frame rates, enabling comprehensive 3D reconstruction of disaster areas.

As a requirement for using the aforementioned camera, the UAV is required to carry an onboard companion computer to process the input, specifically an Nvidia Jetson Family computer is needed to utilize the Stereo Labs camera such as the ZED X. For this specific implementation the Jetson Orin Nano 8GB was used.

2.2 Mapping Algorithm Implementation

Our mapping solution leverages the ZED SDK's SLAM[9] capabilities to create real-time 3D point clouds of disaster areas. Besides real-time depth mapping, the ZED SDK provides positional tracking, and 3D mapping capabilities that work together to extract comprehensive environmental information. The system generates dense point clouds that accurately represent the physical structure of disaster-affected areas, including debris, damaged buildings, and accessible pathways for rescue operations.

The mapping process begins with the UAV following a pre-programmed flight pattern over the designated disaster area. The spatial mapping process runs automatically in the background during image acquisition, continuously updating the 3D mesh as new frames are captured. This approach ensures comprehensive coverage while maintaining computational efficiency suitable for real-time operations.

The route parameters such as camera inclination, velocity and altitude are the result of experimental data obtained from various flights in different conditions such as time of the day and object density, however, these parameters are left open to modification as the conditions in a real life scenario could change drastically, considering the focus on disaster areas

2.3 Data Transmission System

In order to efficiently transmit a large map only the updates or the difference between current state and the newly acquired information is transmitted at any given time, the ZED SDK makes it really easy to stream updates. When configured to do so, the SDK keeps track of the map[5] in sections called chunks and notifies whether or not any particular chunk has been updated.

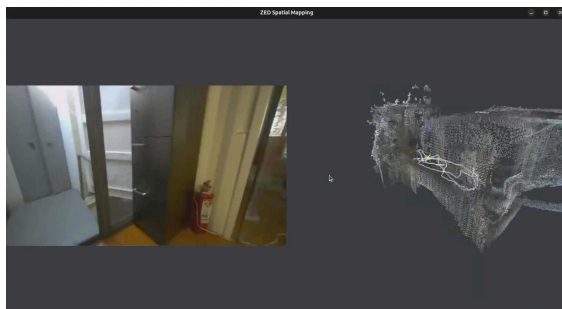


Figure 1: ZED Spatial Mapping example

Taking advantage of the chunk partitioning, the system transmits only the updated chunks. The chunks are cloud points where each point is given by $x, y, z \in \mathbb{R}$ so even a couple of points could represent a substantial amount of bytes. To mitigate the message size problem, a general use compression algorithm is used. The zlib[2] is a free,

general-purpose, legally unencumbered, lossless data-compression library for use on virtually any computer hardware and operating system and for this particular type of data reduces the size of each chunk by approximately 50%, a substantial reduction, allowing the faster transmission required to stream real-time data

As the chunks update with the ZED camera input, the compressed coordinates of each point in the cloud point is sent to a ground station. A 4G LTE cellular modem connected to the companion computer is used for the chunk transmission to the ground station in real-time. The need for such a module comes from the amount of data created by the mapping operation. One explored alternative was to use the FTP capabilities of MAVLink to transmit the points but it was discarded as the majority of telemetry antennas have a very limited bandwidth and the transmission of meaningful information regarding the map was too slow to be considered real-time.

As a final connecting component connecting the onboard companion computer to the ground station a VPN is used. In concrete, the zerotier VPN is used as its UDP hole punching technology offers very low latency and as much bandwidth as each peer has available.

3 PERSON IDENTIFICATION AND LOCATION

3.1 Computer Vision Integration

The person detection and localization system is built around the integration of the ZED stereo camera from Stereolabs, which provides high-resolution stereo imagery along with depth information. By leveraging the ZED SDK body tracking module, the system is capable of tracking multiple individuals simultaneously and extracting precise 3D skeletal and positional information. Unlike conventional 2D vision systems that rely solely on bounding boxes, this approach incorporates depth data to reconstruct three-dimensional coordinates of detected persons, improving both robustness and accuracy in real-world environments.



Figure 2: Object detection bounding boxes on people jogging

The process begins with the UAV performing a structured scanning operation over the designated search area. The UAV advances in discrete intervals, pausing periodically to stabilize its sensors and capture still frames with minimal motion blur. This stop-and-scan strategy ensures that the captured images are not only clearer for human detection but also allow the onboard computer additional processing time in case potential matches are identified.

Captured frames are processed using the ZED SDK's object detection tools, which are specifically configured for person detection mode. For every detection, the system generates bounding boxes around the identified humans and outputs associated relative 3D coordinates with respect to the camera's frame of reference. By coupling these local detections with the UAV's global pose—obtained from onboard navigation sensors and GPS—the system transforms relative coordinates into precise global position estimates of each detected person. This provides a reliable geospatial anchor that can be used later for verification, reporting, and UAV navigation.

Once bounding boxes are extracted, the system crops the relevant regions of interest from the original frame. These cropped images are transmitted via FTP over MAVLink telemetry to the ground control station. This transmission step is crucial because it ensures that search coordinators and AI-based analysis systems can receive lightweight, focused visual data rather than entire high-resolution frames, reducing communication overhead while preserving the details necessary for person identification.

3.2 Natural Language Processing Integration

When the system identifies a possible match, it transitions from the broad scan mode into a focused observation mode. The UAV navigates toward the calculated global coordinates of the suspected individual, maintaining a safe and predefined distance to avoid disturbing or endangering the target. At this closer range, the UAV repeats the detection and image-cropping process.

The cropped image from this secondary observation is again sent to the GPT-4 Vision pipeline, but this time with a higher confidence threshold. The rationale is that images taken at closer range generally contain less distortion and greater detail, which enables more reliable confirmation. Only if the second analysis surpasses this stricter threshold does the system officially confirm that the target person has been found.

If the secondary analysis fails to confirm the target (e.g., due to false positives or environmental artifacts in the initial scan), the UAV resumes its original search route and continues scanning. This layered verification pipeline reduces the likelihood of misclassification, ensuring that the UAV only commits to target engagement when confidence is extremely high.

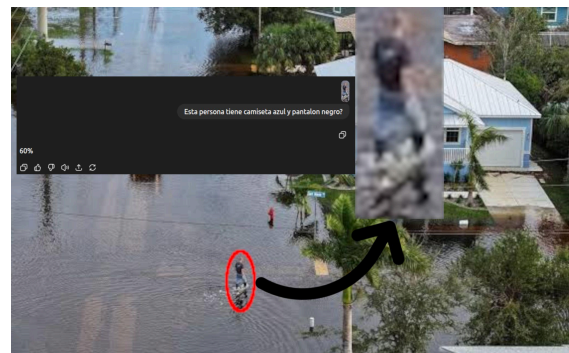


Figure 3: Person zoomed, described with confidence score provided by the LLM

3.3 Search Pattern Optimization

The UAV's search trajectory is governed by an optimized coverage algorithm. The path balances two competing factors: wide-area coverage and sufficient image resolution for reliable detection. The UAV varies its altitude strategically: higher altitudes provide broader scanning coverage, while lower altitudes are used for detailed observation of promising detections. This ensures both efficiency in scanning large areas and precision in confirming individual matches.

The UAV follows a grid-like or spiral pattern, depending on the terrain and mission constraints, with carefully planned pauses to capture stable imagery at each interval. When a detection is confirmed, the UAV abandons the scan temporarily, switching into a target engagement mode as described earlier. If the target is verified, the UAV transitions into the final operational phase.

3.4 Final Engagement and Landing Protocol

Once a target individual has been positively confirmed through the multi-stage verification process, the UAV

initiates a controlled descent to approach the person. Importantly, the UAV is programmed to land at a minimum safe distance of two meters from the detected individual. This buffer zone is essential for safety and regulatory compliance, as any direct collision with the person would not only jeopardize mission integrity but also result in disqualification under competition or operational guidelines.

To ensure the best positioning of the UAV based on GPS coordinates the drone has a high-precision GNSS[11] which communicates with the ground control station which ensures centimeter-level accuracy unlike with conventional GPS systems such as the one found on phones which have around a meter level accuracy.

The descent sequence is carefully managed to avoid obstacles, using both onboard depth sensing and external localization inputs. The UAV ensures that it approaches the person from an angle and trajectory that minimizes risk, while still maintaining visual contact for continuous monitoring until landing is complete.

4 WATER DELIVERY AND RELEASE MECHANISM

4.1 Mechanical Design

The water delivery system employs a bucket suspended beneath the UAV by a 2-meter rope, providing sufficient clearance to avoid interference with the rotor downwash while maintaining a stable and controllable load position.

The release mechanism utilizes a cable retraction system driven by a servomotor mounted on the underside of the UAV, enabling the lifting and release of the bucket's contents at the desired drop point.

The bucket design incorporates a primary attachment point at the top for the suspension rope, as well as a secondary attachment point at the bottom connected to the servo-controlled release mechanism. This dual-attachment approach ensures reliable load retention during flight and allows precise release over target areas.

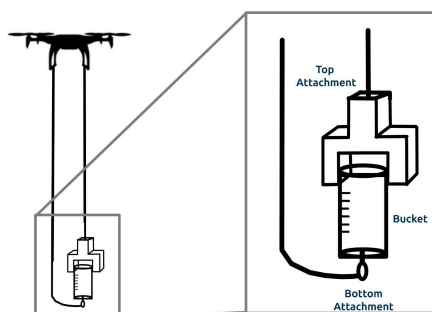


Figure 4: Mechanical Design.

4.2 Servo Control System

The water delivery mechanism requires a reliable release system that can be integrated with the UAV's onboard electronics without introducing mechanical risks or excessive workload to the pilot. To achieve this, a continuous-type servomotor is employed, controlled via Pulse-Width Modulation (PWM) signals.

In conventional hobby-grade servos, the PWM signal is transmitted at a standard frequency of 50 Hz (20 ms period). The pulse width determines the angular position of the shaft: approximately 1.0 ms corresponds to the minimum limit ($\approx 0^\circ$), 1.5 ms to the center point ($\approx 90^\circ$), and 2.0 ms to the maximum limit ($\approx 180^\circ$) [1]. This principle is based on an internal feedback loop, in which a potentiometer coupled to the shaft measures the actual position, while a comparator circuit corrects deviations until the reference value indicated by the pulse is reached[6].

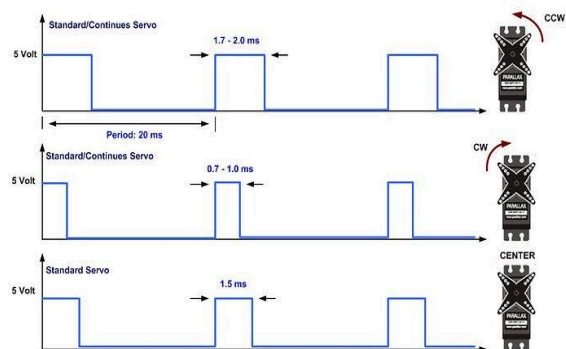


Figure 5: Servo Motor PWM Timing Diagram.

However, not all servos interpret PWM in the same way. Unlike a conventional servo, a continuous-type servo is not designed to hold a specific angular position but rather to rotate continuously, controlling both speed and direction of rotation. They are commonly used in radio-control (RC) applications for winding or unwinding cables. In this type of servo, a 1.5 ms pulse corresponds to a neutral or idle state. Pulses shorter than this value cause rotation in one direction, while longer pulses induce rotation in the opposite direction. The deviation magnitude relative to the neutral value determines the rotational speed.

Servo control is integrated into the flight electronics through a Pixhawk controller. In this environment, PWM values are typically configured between 1000 μ s (minimum), 1500 μ s (neutral), and 2000 μ s (maximum) at a frequency of 50 Hz, allowing the servo to be actuated either manually from the ground control station or through pre-programmed missions in QGroundControl[7]. This operational flexibility is critical in emergency response and assistance environments, where both autonomy and direct operator intervention are required.

4.3 Water Loading Process

Loading. The loading procedure relies on manipulating the bucket angle via the servo-cable system. Initially, the servo retracts the secondary control cable, tilting the bucket to a nearly horizontal (almost inverted) position. In this configuration, the UAV descends to the water supply point, allowing the liquid to easily enter the container.

Once loading is complete, the servo gradually releases the cable tension. By leveraging gravity, the bucket naturally returns to its vertical position, ensuring the contents remain stable during ascent and transport.

Unloading. The unloading process is carried out upon reaching the target zone. The UAV descends to a safe and controlled operational altitude, low enough to guarantee accuracy but avoiding direct interaction between rotor downwash and the surface. The servo then reapplies tension to the secondary cable, tilting the bucket toward the target.

This mechanism enables the directed and controlled release of the entire water load, optimizing delivery accuracy. After discharge is complete, the cable is released again, returning the bucket to its vertical position, ready for the next loading cycle.

4.4 Flight Path Planning and Execution

Mission planning and trajectory execution are managed through QGroundControl software, which enables autonomous missions to be programmed based on predefined GPS coordinates. This environment not only manages navigation waypoints but also actuates the servo, synchronizing the loading and unloading process with UAV positioning[8].

The mission cycle follows these phases:

1. **Takeoff and Initial Ascent:** The UAV lifts off from the designated zone and climbs to a safety altitude that minimizes collision risks with nearby obstacles.
2. **Controlled Descent for Loading:** The UAV navigates to the water supply point and descends to an altitude allowing the bucket, suspended by the rope, to be partially submerged. An approximate residual cable length of 1 meter is considered to ensure proper immersion of the container without compromising UAV stability.
3. **Ascent and Transfer:** After loading is complete, the UAV ascends again to a safety altitude and travels in a straight line toward the delivery point, optimizing energy consumption and minimizing trajectory deviations.
4. **Water Release:** Upon arrival at the target area, the UAV descends to an operational altitude that ensures release precision. At this point, the actuator triggers the bucket mechanism, enabling water discharge.

5. **Cycle Repetition:** The loading and unloading process is continuously repeated along the programmed route until battery levels reach the safety threshold.
6. **Automatic Return:** Once this threshold is reached, the UAV activates the Return to Launch (RTL) function, autonomously returning to the takeoff zone for a safe landing.

This procedure ensures a balance between autonomy and delivery precision, enabling multiple water transport cycles with minimal operator intervention.

5 GENERAL SYSTEM SPECIFICATION

The document would not be complete without the proper description of the components used in the UAV and its relevance to the challenges solutions.

5.1 The software

The software architecture for the system is a distributed series of nodes with specific tasks and supporting the communication between each node is a ROS[13] inspired system called forerunner2[12]. Even though the Robot Operating System is widely used, it has some limitations, like the platforms it is easy to compile against are constrained and the size required for a full installation is considerable.

To ease the downsides of ROS, a simpler yet useful system was developed. In it nodes still communicate through a publisher-subscriber model, with the addition of actions, but with a more direct approach removing the overhead of distributed discoverability and replaced by a centralized registry that keeps track of the node address while maintaining a direct peer to peer communication. The next diagram shows how each node establishes a connection

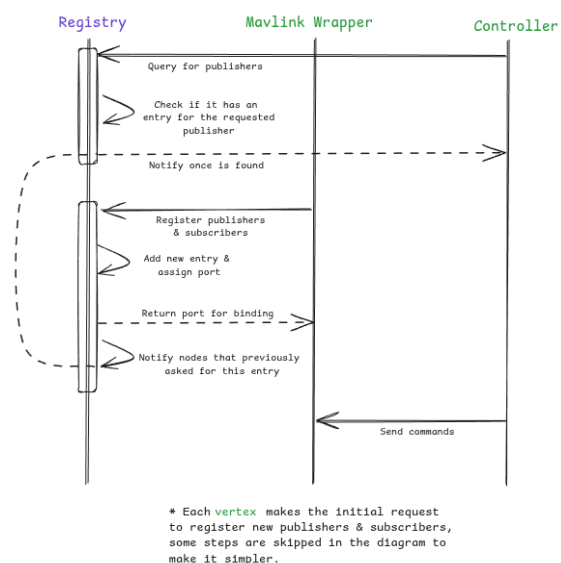


Figure 6: Node Communication

For establishing a communication channel between peers, the ZMQ[14] library is used, which allows granular control and very little overhead while maintaining the code simplicity. The final piece of the puzzle is the serialization library, the CapNProto library which is faster than other solutions (e.g. Protobuf[15]).

The custom made nature of the communication platform keeps the code really simple and attached to the necessities at hand and removes the unnecessary tools included with other solutions.

Another important piece of software used and integrated into our current stack of development is Rerun[17], which allows us to have a visualizer for the different messages being sent across our system that makes it easier to debug, replay previous flights and get real time visual feedback from missions.

The autopilot

A key piece for any Autonomous Vehicle is the software responsible for taking the input or desired state and output the control commands to the hardware. PX4[16] was used as an intermediary between the desired trajectory and the commands sent to the rotors.

The capabilities of an autopilot software are beyond just a control unit as it allows the communication to the ground stations, the use of onboard computers and the inclusion of multiple sensors to the system, making them available not just for the internal calculations but also exposing them to other components in need

5.2 The quadrotor

The aerial vehicle proposed for the solution is based on the Holybro x650 Developer kit platform with some additional sensors like an RTK system, a downward facing optical camera and a more powerful telemetry antenna. The next table contains all the components included in the vehicle

Part	Component
FMU	Pixhawk 6x
Frame	X650
Motor	Tmotor MN4014 KV330
ESC	Tekko32 F4 45A
Camera	ZED X
Telemetry	Microhard Telemetry Radio P900

Power distribution	Holybro PDB01-V1.2 & PM06D
GNSS/RTK	H-RTK F9P
Propellers	Gemfan 1555 Carbon Fiber
Radio Controller	TX16S Mark II

Table 1: UAV Components

This combination of components allows for high accuracy flights from waypoints and a sufficient distance of control through the telemetry antenna, making the resulting kit ideal for disaster operations as it can travel long distances and report its position or other markers, like persons in need of rescue, with high accuracy.

5.3 The companion computer

One of the biggest disadvantages of using a robust visual solution like the ZED camera is the need of an onboard computer to process the image. And not any computer, specifically an Nvidia Jetson[18] platform, which increases the cost and greatly reduces the battery autonomy.

In spite of the problems of carrying a companion computer with the dimensions and power consumption of an Nvidia Jetson, the precision and capabilities of the ZED X camera greatly surpasses the negative aspects.

So the focus was to minimize the impact of the ZED camera in the drone, and while doing so, the Jetson Orin Nano was the best choice. The cheapest computer of the Nvidia Jetson lineup is also the least power consuming while still being able to run the ZED SDK without any issue. The only drawback is the lack of support for hardware video encoding.

The companion computer will communicate with the autopilot through a UART interface and the MAVLink protocol, which enables a bidirectional communication channel. Where the Jetson Nano can listen to events, read sensor data, read the current mission while also send control commands to the PX4 software in order to make a coordinated search for the mapping and search challenges

For the mapping challenge the companion computer is also going to be carrying a 4G cellular modem in order to transmit in real-time the cloudpoint to the ground station.

Additionally a step down regulator had to be mounted and connected to the PDB in order to provide the right voltage required by the Jetson computer.

5.4 The Ground Station

The ground station is the combination of hardware and software that allows the communication and control with

the Aerial Vehicle. For this proposal, the ground stations will also be receiving information from a companion computer in order to render a cloud point map of a designated area.

The software from the ground station should know in advance how to communicate with the vehicle firmware. In respect to PX4 there is already a very powerful program that can do mission planning, actuator activation and monitor. The QGroundControl software is an open source project aimed to the PX4 flight controller and allows seamless integration with any aerial vehicle that supports the PX4 firmware

6 PUBLISHING POLICY

The IMAV proceedings will only be distributed in digital form. The submitting author is responsible for obtaining agreement of all his / her coauthors and any consent required from sponsors before submitting a paper. The authors are obliged to cite relevant prior work.

7 CONCLUSION

The system presented in this work constitutes a fully autonomous UAV solution that effectively addresses the main challenges encountered in disaster response operations through the integration of three fundamental mission modules: three-dimensional mapping of affected areas, person identification and localization, and a reliable water delivery mechanism. Each of these components was implemented and validated within a single aerial platform, demonstrating that a high level of autonomy and precision can be achieved without the need for multiple specialized vehicles.

The ZED X stereo camera enabled the generation of precise and continuously updated three-dimensional models, providing essential situational awareness for terrain assessment. The integration of computer vision with natural language processing through OpenAI's API allowed reliable person identification from verbal descriptions, demonstrating effectiveness in search and rescue scenarios. Finally, the servo-controlled water release mechanism achieved stable and accurate performance, ensuring precise resource delivery under critical conditions.

Taken as a whole, the developed UAV represents a coherent technical solution that successfully integrates advanced perception technologies, artificial intelligence algorithms, and reliable actuation mechanisms within a single compact aerial platform. The results validate the feasibility of performing accurate three-dimensional mapping, robust person identification, and precise resource delivery with one vehicle, consolidating an

effective and versatile approach for emergency scenarios. This conclusion emphasizes that by focusing on the unification of specialized subsystems under a single operational framework, it is possible to design autonomous aerial solutions that maximize response capability while minimizing dependence on external infrastructure during critical missions.

ACKNOWLEDGEMENTS

The authors acknowledge the support of the International Micro Air Vehicle Competition organizers and the valuable contributions of the open-source robotics community in developing the underlying technologies that made this work possible.

REFERENCES

- [1] Stereolabs. ZED X - AI Stereo Camera for Robotics. Available online: <https://www.stereolabs.com/products/zed-x> (accessed on 28 August 2025).
- [2] (N.d.). Zlib.net. Retrieved August 29, 2025, from <https://zlib.net/>
- [3] *Mavsdk::FtpServer class reference*. (n.d.). Mavlink.Io. Retrieved August 29, 2025, from https://mavsdk.mavlink.io/main/en/cpp/api_reference/classmavsdk_1_1_ftp_server.html
- [4] *Holybro*. (n.d.). Holybro.com. Retrieved August 29, 2025, from <https://docs.holybro.com/radio/microhard-radio/specification>
- [5] Stereolabs Medium Blog. Spatial Mapping in Computer vision using ZED. Available online: <https://stereolabs.medium.com/spatial-mapping-in-computer-vision-using-zed-69bce43c2e7a> (accessed on 28 August 2025).
- [6] "Servo control," Wikipedia, 2025. [Online]. Available: https://en.wikipedia.org/wiki/Servo_control
- [7] "Actuators — PX4 Developer Guide," PX4 Docs, 2025. [Online]. Available: <https://docs.px4.io/main/en/config/actuators.html>
- [8] "Plan View — QGroundControl User Guide," QGroundControl Docs, 2025. [Online]. Available:

https://docs.qgroundcontrol.com/master/en/qgc-user-guide/plan_view/plan_view.html

[9] (PDF) The Simultaneous Localization and Mapping (SLAM)-An Overview. (n.d.). *ResearchGate*. <https://doi.org/10.38094/jastt204117>

[10] *Depth Modes - Stereolabs*. (n.d.). <https://www.stereolabs.com/docs/depth-sensing/depth-modes>

[11]: “High precision positioning,” u-blox, Nov. 24, 2021. <https://www.u-blox.com/en/technologies/high-precision-positioning>

[12]: covenant-org, “GitHub - covenant-org/forerunner2,” GitHub, 2025. <https://github.com/covenant-org/forerunner2> (accessed Sep. 05, 2025).

[13]: “ROS 2 Documentation — ROS 2 Documentation: Kilted documentation,” Ros.org, 2025. <https://docs.ros.org/en/kilted/index.html> (accessed Sep. 05, 2025).

[14]: Zeromq.org, Sep. 07, 2023. <https://zguide.zeromq.org/docs/preface/> (accessed Sep. 05, 2025).

[15]: “Overview,” Protocol Buffers Documentation. <https://protobuf.dev/overview/>

[16]: “PX4 User Guide,” docs.px4.io. <https://docs.px4.io/main/en/>

[17]: “What is Rerun?,” Rerun.io, 2025. <https://rerun.io/docs/getting-started/what-is-rerun> (accessed Sep. 05, 2025).

[18]: “Jetson Orin Nano Super Developer Kit,” NVIDIA, 2025. <https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-orin/nano-super-developer-kit/>